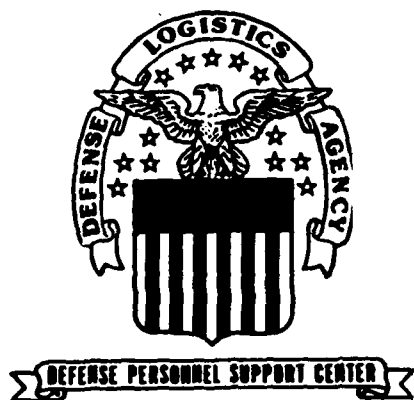


2



AD-A223 222

MEDICAL SIMULATION MODEL

TO COMPARE

SAMMS VS ARQ

INVENTORY POLICY

DTIC
ELECTE
JUN 26 1990
S D D

FINAL REPORT

JUNE 1990

DISTRIBUTION STATEMENT A

**Approved for public release
Distribution Unlimited**

**OPERATIONS RESEARCH AND
ECONOMIC ANALYSIS OFFICE**

OFFICE OF POLICY AND PLANS

**DEFENSE PERSONNEL SUPPORT CENTER
2800 South 20th Street, Philadelphia, PA 19101-8419**

90 06 25 190

MEDICAL SIMULATION MODEL TO COMPARE
SAMMS INVENTORY POLICY WITH THE
AVERAGE REQUISITION QUANTITY INVENTORY POLICY

FINAL REPORT
JUNE 1990

Francis A. Daniels
Office of Policy and Plans
Operations Research and Economic Analysis Office
Defense Personnel Support Center
Philadelphia, Pennsylvania

CONTENTS

<u>Title</u>	<u>Page</u>
Contents.....	i
List of Figures.....	iii
Executive Summary.....	iv
I. Introduction.....	1
A. Background.....	1
B. Problem Statement.....	1
C. Objectives.....	2
II. Methodology.....	2
A. Current SAMMS Processes and Methodologies.....	2
(1) Uniform Materiel Movement and Issue Priority.....	
System (UMMIPS).....	2
(2) The Maximum Release Quantity (MRQ).....	3
(3) Requirements Objective Calculation.....	3
(4) The Procurement of Stock.....	4
(5) Demand Forecasting.....	4
(6) How the System Fills Requisitions.....	5
(7) Creation of Backorders.....	5
B. The Average Requisition Quantity (ARQ) Process.....	6
C. The Simulation Model.....	6
(1) Data Extraction.....	6
(2) Data Filtering.....	7
(3) Creation of Files With NSNs to be Considered For... Simulation.....	8
(4) Random Selection of Items For Analysis.....	9
(5) Calculation of The Number of Requisitions Per Item..	9
(6) Determination of An Item's Average Requisition..... Quantity.....	9
(7) Requisition Generation--Quantity In Relation to... ARQ.....	10
(8) Probability of A Duein Being Late.....	10
(9) Other System Parameters.....	12
(10) Assumptions.....	12
(A) Simulated Inventory Policy.....	12
(B) Demand Forecasting.....	12
(C) Generation/Filling of Backorders.....	14
(1) Current SAMMS Process.....	14
(2) The ARQ Process.....	14
(3) The Filling of Backorders.....	14

III.	Analysis.....	14
IV.	Conclusions.....	15
V.	Recommendations.....	15
VI.	Benefits.....	16
Appendices.....		
Appendix A	Documentation for the Variables Used in the.. Medical Simulation Model.....	A-1
Appendix B	Computer Listing of the SAMMS Simulation..... Portion of the Model.....	B-1
Appendix C	Computer Listing of the ARQ Simulation..... Portion of the Model.....	C-1
Appendix D	Sensitivity Analysis Results.....	D-1

LIST OF FIGURES

<u>Number</u>	<u>Title</u>	<u>Page</u>
1	Basic ARQ Concept--FY 88 Medical Data	11
2	Probability Distribution of a Duein Being Late	13

EXECUTIVE SUMMARY

Over the past several years, the Directorate of Medical Materiel has been experiencing an increase in the number of backorders established, and a corresponding reduction in supply availability. To alleviate the problem, the Directorate requested that Headquarters, DLA authorize a test of a requisition policy that 'optimizes' the issue of assets for items at or near a critical stock position. This policy, called the Requisition Optimization concept was previously used by Medical, but discontinued in 1983 because of policy conflicts with the Uniform Materiel Movement and Issue Priority System (UMMIPS). Under the optimization policy, supply availability was approximately 95%. When the policy was discontinued, supply availability began to decrease, and subsequently fell to below 90%.

The Requisition Optimization Quantity concept is essentially a modified maximum release quantity concept. It looks at issue priority group two (2) and three (3) requisitions and predetermines which will be filled and which will automatically go on backorder. IPG 1 requisitions are treated exactly as in SAMMS. The optimization concept rations the stock of items in a critical position, i.e., less than 30 days of issuable stock, by putting larger requisitions on backorder and utilizing that stock to fill more smaller requisitions. The size of the IPG 2 and 3 requisitions that are automatically backordered is determined by looking at their size in relation to the average size of a requisition received for that NSN. Currently IPG 2 requisitions have an optimization hold whenever the requisition calls for a quantity greater than 300% of the average requisition size. IPG 3 requisitions have an optimization hold whenever the requisition is greater than 150% of the average requisition size for the specific NSN.

This model simulates the SAMMS system in Medical and the optimization process. It looks at the results over a five year time period and determines if the optimization process increases supply availability and decreases backorders. Results indicate that the optimization process, over time, is neither better nor worse than the SAMMS system. Rather than produce a consistent increase in supply availability and decrease in backorders, the optimization process behaves like an insurance policy that is there if needed. This insurance comes into play when the critical stock position has been breached and expected due ins are delayed. In this scenario the optimization process is useful; however, if the due ins are imminent, then backorders will be unnecessarily created. Over the long run the trade off is about equal. In the short run, when contracting problems impact expected due in delivery schedules, the optimization policy will be beneficial. The key to use of an optimization policy is the ability to identify when that policy should be invoked.

It is recommended that: 1) Medical continue using the optimization concept under the following conditions: the optimization trigger be set at a 45 day stock level, that IPG 2 and IPG 3 requisitions be backordered when the requisition is 2 times an item's average requisition quantity, and that FSC 6540, optical lenses and equipment not be included; and 3) Medical review the SAMMS National Inventory Record and Supply Control Files for updates and corrections to data elements currently being utilized in SAMMS.

REQUISITION OPTIMIZATION MODEL

I. INTRODUCTION

This report has been prepared for the Directorate of Medical Materiel in response to their request to 'provide the necessary technical or statistical information through simulation' to justify the implementation of the Requisition Optimization inventory policy.

A. BACKGROUND

Over the past several years, the Directorate of Medical Materiel has been experiencing an increase in the number of backorders established, and a corresponding reduction in supply availability. To alleviate the problem, the Directorate requested that Headquarters, DLA authorize a test of a requisition policy that 'optimizes' the issue of assets for items at or near a critical stock position. This policy, called the Average Requisition Quantity (ARQ) concept was previously used by Medical, but discontinued in 1983 because of policy conflicts with the Uniform Materiel Movement and Issue Priority System (UMMIPS). Subsequent to the discontinuance of the ARQ policy, supply availability began to decrease falling to levels below 90%. At least part of the fall in supply availability was attributed to the removal of the ARQ from Medical inventory management policies. With the 1989 request to DLA to reestablish ARQ, Medical anticipated that ARQ would increase supply availability substantially.

The Average Requisition Quantity (ARQ) concept is essentially a modified maximum release quantity (MRQ) concept. It looks at issue priority group (IPG) two (2) and three (3) requisitions and predetermines which will be filled and which will go on forced backorder. IPG one (1) requisitions are filled as they normally would be. The concept attempts to ration items in critical stock positions by filling more smaller requisitions while placing the larger ones on backorder. Requisitions will be force backordered, once a critical stock position is passed, whenever their size is 300% of the average requisition size for the NSN for an IPG 2 or 150% of the average requisition size for an IPG 3. Cut off points are empirically determined by analyzing the cumulative frequencies of requisitions and demands received for that item.

B. PROBLEM STATEMENT

As stated in the Project Study Plan provided Medical (February 89), the purpose of the study would be to develop a specialized simulation model to test the effectiveness of the ARQ inventory policy. The simulation developed is a Monte Carlo model that looks at the interactive dynamics of supply operations in the Medical Directorate over a long period of time to determine the possible affects of implementing a new (ARQ) inventory policy, and compare its results with the SAMMS system as it currently operates. Because of the dynamics that characterize a sophisticated inventory operation, any proposed policy change must be looked at over the long run.

Basing projections of ARQ success on an analysis of how the system would react to a specific asset position, backorder scenario, and static estimates of ALT/PLT was inadvisable as any results, positive or negative, may not be indicative of the expected impact over the long run. Therefore, simulation was chosen to allow for the dynamics of the system to interact to the policy change, drive the system for a sufficient period of time, and allow for the collection of a data that would better represent an analysis of the results that will probably be obtained.

C. OBJECTIVES

The objectives of the study were twofold. First, was to evaluate/measure the effectiveness of implementing the ARQ inventory release policy. The measure of this increase in effectiveness would be the increase in supply availability, and the corresponding decrease in backorders. Secondly, once this model was constructed, use it as a starting point for the development of a generalized model of Medical's supply operation to be utilized in further studies.

II. METHODOLOGY

A. CURRENT SAMMS PROCESSES AND METHODOLOGIES

The Supply Operations Division of the Medical Directorate is responsible for satisfying customer requisitions. To accomplish this mission, the Standard Automated Materiel Management System (SAMMS) is utilized. DLAM 4140.2, Volume II, Part 1, Chapter 4, Requisition/Issue transactions outlines the procedures 'governing the receipt and processing of MILSTRIP (Military Standard Requisitioning and Issue Procedures) requisitions'. In order to control the issuance of stock, and still meet customer demands, the requisitioning process requires customers to submit requisitions coded with a specific priority. These priorities, known as Issue Priority Groups are broken down into three major categories, each having its own subcategories. Requisitions are submitted as IPG ones (1's), twos (2's), and threes (3's), with IPG 1 being the highest priority and IPG 3 the lowest. When reference is made in this paper to 'high priority', IPG 1 requisitions are being discussed. In SAMMS, the Uniform Materiel Movement and Issue Priority System (UMMIPS) level and a Maximum Release Quantity (MRQ) in conjunction with the IPG concept are used to prevent the issuance of an excessively large amount of stock. These parameters can be overridden by an item manager, but they are the basic stopgaps used to insure that stock on hand is issued within a specified quantity parameter.

(1) Uniform Materiel Movement and Issue Priority System (UMMIPS)

The UMMIPS criteria can be found in DLAM 4140.2, Volume II, Part 2, Appendix D-173. It states that a control level is established for each item "by taking given percentages of specified data to determine that quantity of an item which is to be reserved for issue to meet high priority when there is no reasonable assurance that additional stocks will be received prior to reaching an out-of-stock position."

The UMMIPS control is designed to set aside stock for IPG 1 requisitions when an item gets into a critical stock position. The current UMMIPS level for the Directorate of Medical is the lesser of 80% of the Safety Level or one-sixth of the quarterly forecast of demand (QFD). For an item forecast quarterly, one-sixth represents approximately fifteen (15) days of stock. For those items forecast monthly, one-sixth represents five (5) days. As the Medical Directorate utilizes a variable safety level for its items, the 80% level will vary. However, according to DLAM 4140.2, Volume II, Part 2, Appendix D-187, the variable safety level for a item is three standard deviations of the leadtime, expressed as a quantity. Should the level be greater than the leadtime, it is adjusted to equal the leadtime. If it is less than the leadtime, it is unchanged. Therefore, the UMMIPS level will vary from item to item, and is reviewed everytime a new forecast is made.

During the process of filling requisitions, stock is issued until on-hand balances reach the UMMIPS level. When this occurs, another review is conducted to determine the type of requisition submitted. If it is an IPG 2 or 3, the requisition is put on back order.

(2) The Maximum Release Quantity (MRQ)

To insure that very large quantities of an item are not released, a maximum release quantity is entered into the SAMMS system. The calculation for the maximum release quantity is found in DLAM 4140.2, Volume II, Part 2, Appendix D-172. It is determined by multiplying the quarterly forecasted demand by a maximum release quantity factor. This factor is found in DLAM 4140.2, Volume II, Part 3, Appendix A-85, and is based on the dollar value forecast of an item's annual demand. Demand is categorized as high (greater than \$4500/year), medium (\$400 - \$4500/year), and low (less than \$400/year). These factors appear in Management Policy Table 018, and are three tenths (.3) for items classified as either having high or medium demands, and one (1.0) for items classified as having low demand. Therefore, the maximum release quantity for an item with a high dollar value forecast of annual demand is .3 times the item's QFD. When a requisition is submitted, its quantity checked against the item's MRQ. As long as the quantity is less than the MRQ, and the UMMIPS level is not violated, the stock is released and the requisition satisfied. A violation against either check creates a partial fill, with the remainder back ordered.

(3) Requirements Objective Calculation

In determining what an item's requirements objective are, the following elements are utilized in the calculation: the item's war reserve protectable level, special program requirements (usually DEPMEDS), the safety level, backorders, the procurement leadtime in months (expressed as a quantity), and the procurement cycle period (expressed as a quantity). The leadtime represents the time period from the time a decision is made to procure an item, to the time it is received in the depot. The cycle represents the stock required between procurements.

(4) The Procurement of Stock

A decision to procure stock is made when on-hand issuable assets plus the stock due in is less than the reorder point. The amount of stock stock procurred is then based on an economic order quantity (EOQ) model outlined in DLAM 4140.2, Volume II, Part 2, Appendix D-186. The EOQ computation is a function of the dollar value of the quarterly demand of an item, its shelf-life, and a constant called the T-factor. The dollar value of the quarterly demand is determined by multiplying the quarterly forecast of demand by the items unit price. The actual EOQ calculation is only used when the dollar value of the quarterly demand is greater than \$62. but less than \$1846. These parmeters are set by the Medical Directorate in conjunction with DLA, and are found in Management Policy Table 018. The calculation is defined as follows:

$$\begin{aligned}\text{EOQ} &= (\text{TFACFOR} * (\text{SQRT}(\text{DVQD}))) \\ \text{MONTHS} &= (\text{EOQ}/\text{UP})/(\text{QFD}/3.)\end{aligned}$$

where:

EOQ = Economic Order Quantity
TFACFOR = A constant utilized in the calculation
SQRT = The square root function
MONTHS = The conversion of the EOQ calculation into a shelf-life equivalent
UP = The item's unit price
QFD = The item's quarterly forecast of demand

If the EOQ calculation is utilized, the actual amount of stock procurred depends on the item's shelf-life. If it is less than the figure determined by the variable MONTHS in the EOQ calculation, and the shelf-life is zero (examples of items not having a shelf life are surgical instruments, hospital equipment, and laboratory equipment), the amount of stock procurred is the item's procurement cycle quantity. If it is less that the figure determined by MONTHS, and the item has a shelf-life (drugs), then the amount of stock procurred is given as follows:

$$\text{DUEIN} = (\text{EOQ}/\text{UP}) * (\text{SHELF-LIFE}/\text{MONTHS})$$

If the item's shelf-life is greater than the figure determined by MONTHS, then the amount of stock procurred is given as follows:

$$\text{DUEIN} = \text{EOQ}/\text{UP}$$

When the EOQ calculation is not utilized, the amount of stock procurred remains a function of the dollar value of the quarterly demand and shelf-life. If the shelf-life is zero, the procurement cycle quantity is bought. If the shelf-life is greater than zero, then a multiple of the quarterly forecast of demand is bought.

(5) Demand Forecasting

The current process utilized by the SAMMS procedure to forecast

customer demand is double exponential smoothing. Forecasting is done on a monthly basis for VIP items, and quarterly for all others. DLAM 4140.2, Volume II, Part 1, chapter 53 discusses the demand forecasting procedure and how it is implemented. To forecast a demand, the SAMMS system uses a single exponential smoothed average to compute the double exponential smoothed average. The actual forecast of demand is then two times the single minus the double. The equations utilized in the forecasting process are as follows:

The single smoothed average calculation is given as:

$$S_t = \alpha(P_t - S_{t-1}) + S_{t-1}$$

where:

S_t = The single smoothed demand
 α = The alpha factor
 P_t = The present period's demand (its current QFD)
 S_{t-1} = The last period's single demand (former QFD)

The double exponential smoothing calculation is given as:

$$D_t = \alpha(S_t - D_{t-1}) + D_{t-1}$$

where:

D_t = The double smoothed demand
 D_{t-1} = The last period's double smoothed demand

The expected demand, i.e, new quarterly forecasted demand then becomes:

$$ED = 2(S_t) - D_t$$

(6) How the System Fills Requisitions

The current SAMMS process to fill a requisition follows a pattern of determining its issue priority group (IPG) (See paragraph II.A. above for a discussion of the Issue Priority Group), the size of the requisition (See paragraph II.A.(2) above for a discussion of the maximum release quantity), and the current on hand inventory balance. Requisitions are filled in terms of time spent on back order, by IPG; and then by date of receipt, by IPG. Therefore, IPG 1 requisitions are always looked at first, and then IPG 2 and 3. A materiel release order (MRO) is the document forwarded to the shipping depot by Medical to fill a requisition. Requisitions are filled (MROs forwarded) as long as the on-hand asset level remains above the UMMIPS level. When the UMMIPS level is breached, back orders are then generated. See paragraph II.A.(1) for a discussion of UMMIPS levels and their corresponding breach levels.

(7) Creation of Back Orders

When one of the UMMIPS breach levels (Paragraph II.A.(1)) has been met, the process of creating back orders is begun. An item will stay

on back order until sufficient stock has been received to satisfy the requisition. Back orders are also established when a requisition violates the item's MRQ. A partial fill is generated, up to the quantity of the MRQ, with the remainder going on back order.

B. THE AVERAGE REQUISITION QUANTITY (ARQ) PROCESS

As stated in the Background, the Average Requisition Quantity (ARQ) concept is essentially a modified maximum release quantity (MRQ) concept that is applied whenever a critical asset position is reached. It treats IPG 1 requisitions the same as the MRQ process, but attempts to ration remaining stock between IPG 2 and IPG 3 requisitions. It accomplishes this by determining which requisitions will automatically go on backorder depending upon how large the requisition is as compared to the average requisition quantity of the NSN. The concept is to fill more smaller requisitions (in relation to the NSN's average requisition size), and put the larger ones on back order. In this manner, the number of requisitions on backorder should decrease. The current Medical policy is that predetermination triggers should be set at three times (3X) the ARQ for IPG 2 requisitions, and one and a half times (1.5X) the ARQ for IPG 3 requisitions.

C. THE SIMULATION MODEL

(1) Data Extraction

The model simulates Medical's supply system and is built in a modular fashion to allow for the modification of its various segments without changing the simulation itself. It utilizes as its base a two year demand history for each NSN used in the simulation. The data provided is for stocked items, supply status code (SSC) one and seven. See paragraph II.C.(4) below for a discussion of how NSN's were chosen for the simulation. The data base utilized by the model consists of data extracted from SAMMS data bases (the National Inventory Record (NIR) and the Supply Control File(SCF)) already in use by Medical. The raw data from these SAMMS files is consolidated into one data base, and for each item consists of the following:

- 1) NSN
- 2) Nomenclature (Item description)
- 3) Unit of Issue
- 4) Case Weight and Cube
- 5) Shelf-Life Code
- 6) Unit Price
- 7) Maximum Release Quantity (MRQ)
- 8) Item Category Code (ICC)

- 9) Reorder Point Quantity
- 10) On Hand Issuable Assets
- 11) VIP Code
- 12) Average Requisition Quantity (ARQ)
- 13) Quarterly Forecasted Demand (QFD)
- 14) System Single Exponential Smoothed Quantity
- 15) System Double Exponential Smoothed Quantity
- 16) Safety Level Quantity
- 17) Alpha Factor
- 18) Lead Time-Days
- 19) Procurement Cycle-Months
- 20) Procurement Level-Months
- 21) War Reserve Requirements
- 22) Special Program Requirements-DEPMEDS
- 23) Recurring Demand with number of requisitions
- 24) Non-recurring Demand with number of requisitions

The above twenty-four data elements are then filtered (See II.C.(2) below) and rewritten to two files that become the source files for items that will be randomly chosen for the simulation.

(2) Data Filtering

Prior to determining which NSN's will be considered for analysis by the model, the raw data base provided by OTIS is reviewed and any NSN's, both VIP and non VIP, not having a quarterly forecast demand, missing an assigned NSN or nomenclature, or not having at least nine (9) months of data are eliminated from consideration. In addition, any items that have migrated to other than a supply status code one or item category code one are also eliminated. Once an item is considered eligible for consideration, an administrative lead time quantity and a procurement leadtime quantity is calculated and stored for use in the data files used by model. Also, in calculating an average requisition quantity, months in which no demands were received were not used in the calculation because of their adverse effect on the average. (See II.C.(5) & (6) below).

Numerous problems were encountered with the data during the filtering process. A synopsis of those problems is provided above,

but others also were also encountered. Alpha factors were not found for the majority of items, and had to be added when an item was considered eligible, and the average requisition quantity on the NIR was not being updated, and could not be used. These problems with the NIR and SCF were a cause for concern and were discussed with Medical personnel.

(3) Creation Of Files With NSNs To Be Considered For Simulation

Once the raw data file provided by OTIS has been filtered, the following two files are created and become the source files from which NSN's are randomly chosen for simulation by the model. Their contents are as follows:

FILE 1

- 1) NSN
- 2) Nomenclature
- 3) VIP Code
- 4) Average Requisition Quantity (ARQ) From NIR
- 5) Quarterly Forecasted Demand QFD (From SCF)
- 6) System Single Exponential Figure
- 7) System Double Exponential Figure
- 8) Safety Level
- 9) Alpha factor
- 10) Admin Lead Time (Days)
- 11) Proc Lead Time (Days)
- 12) Proc Cycle Period in Months
- 13) Shelf-Life
- 14) MAD (Mean Absolute Deviation)

FILE 2

- 1) NSN
- 2) Unit Price
- 3) Maximum Release Quantity (MRQ)
- 4) Item Category Code (ICC)
- 5) Reorder Point (ROP)
- 6) On-Hand Assets (OHA)
- 7) Special Program Req'ts. (SPR)
- 8) War Reserve Level (WRL)
- 9) # Requisitions Rec'd in last 24 months
- 10) Calculated ARQ
- 11) # of months of demand in last 24
- 12) Proc Cycle Period-Quantity
- 13) Admin/Proc Leadtime Quantity
- 14) Proc Leadtime Months

(4) Random Selection Of Items For Analysis

A sample of approximately 10% of the NSNs considered eligible for simulation was drawn. This equated to approximately 878 items.

(5) Calculation Of The Number Of Requisitions Per Item

Given the scope and breath of items managed by Medical, the simulation must decide how many requisitions to generate for each NSN being analyzed. This is done by calculating for each NSN how many requisitions per day, or how many days pass between receipt of requisitions, from the distribution of requisitions received for the NSN during the preceeding twenty-four months. If an item has received, on average, eight (8) requisitions per day for the past 24 months, it will continue to do so during the simulation. Conversely, if an item, on average, has received a requisition every five (5) days, then it will only get a requisition every fifth day during the simulation. The calculations to determine the number of requisitions per day, or how many days pass between the receipt of a requisition for each NSN are as follows:

$$\begin{aligned}RM &= \text{SUM1}/\text{INUM1} \\RD &= RM/30. \\LM &= \text{INT}(1./RD) + .5\end{aligned}$$

where:

RM = The number of requisitions received per month (for the number of months requisitions were received).

SUM1 = The actual number of requisitions received during the past twenty four months, for the number of months that requisitions were received.

INUM1 = The number of months that requisitions were received during the past twenty four months.

RD = The number of requisitions per day. The model uses a 30 day month, as the Directorate receives requisitions every day of the year.

If RD is less than one, then LM determines how many days pass between the receipt of requisitions.

If RD is greater than one, then a variable named IREQS1 determines how many requisitions per day an NSN will receive.

(6) Determination Of An Item's Average Requisition Quantity (ARQ)

The average requisition quantity, (See paragraph II.C.(1), item

12) provided from the National Inventory Record (NIR) was not utilized because the NIR figure is not updated to reflect current requisition activity. An ARQ is calculated for each item analyzed by the system (See paragraph II.C.(3), item 10), utilizing the following:

$$ATOT = TOT/SUM$$

where:

ATOT = The average requisition size for the NSN being analyzed.

TOT = The total quantity requisitioned during the past twenty four months.

SUM = The number of requisitions received during the past twenty four months.

(7) Requisition Generation--Quantity In Relation To ARQ

Individual requisition quantities are simulated by the model based upon an analysis of FY 88 Medical requisitions contained in the DLA Integrated Data Bank. To obtain these requisitions, a random selection of 10% of the status code 1 and 7 items was made. This produced a sample of one thousand, two hundred and seventy-nine (1,279) NSN's. The average requisition quantity for each NSN chosen was calculated for FY 88. Then, the NSN's were matched against all requisitions received for the 4th quarter FY 88. This matching produced 41,796 requisitions to be used in developing the probability distribution of ARQ multiples with their corresponding demands for use by the model. The 41,796 requisitions produced 4,226,184 units of demand.

This data produced figure 1 which shows that 73.5% of the requisitions received (30,706) were for an ARQ multiple of one, producing 16.7% of the total demand (706,507 units). At the other extreme, .66% of the requisitions received (276) were of an ARQ multiple of sixteen (16) and produced 29.7% of the total demand (1,254,177 units).

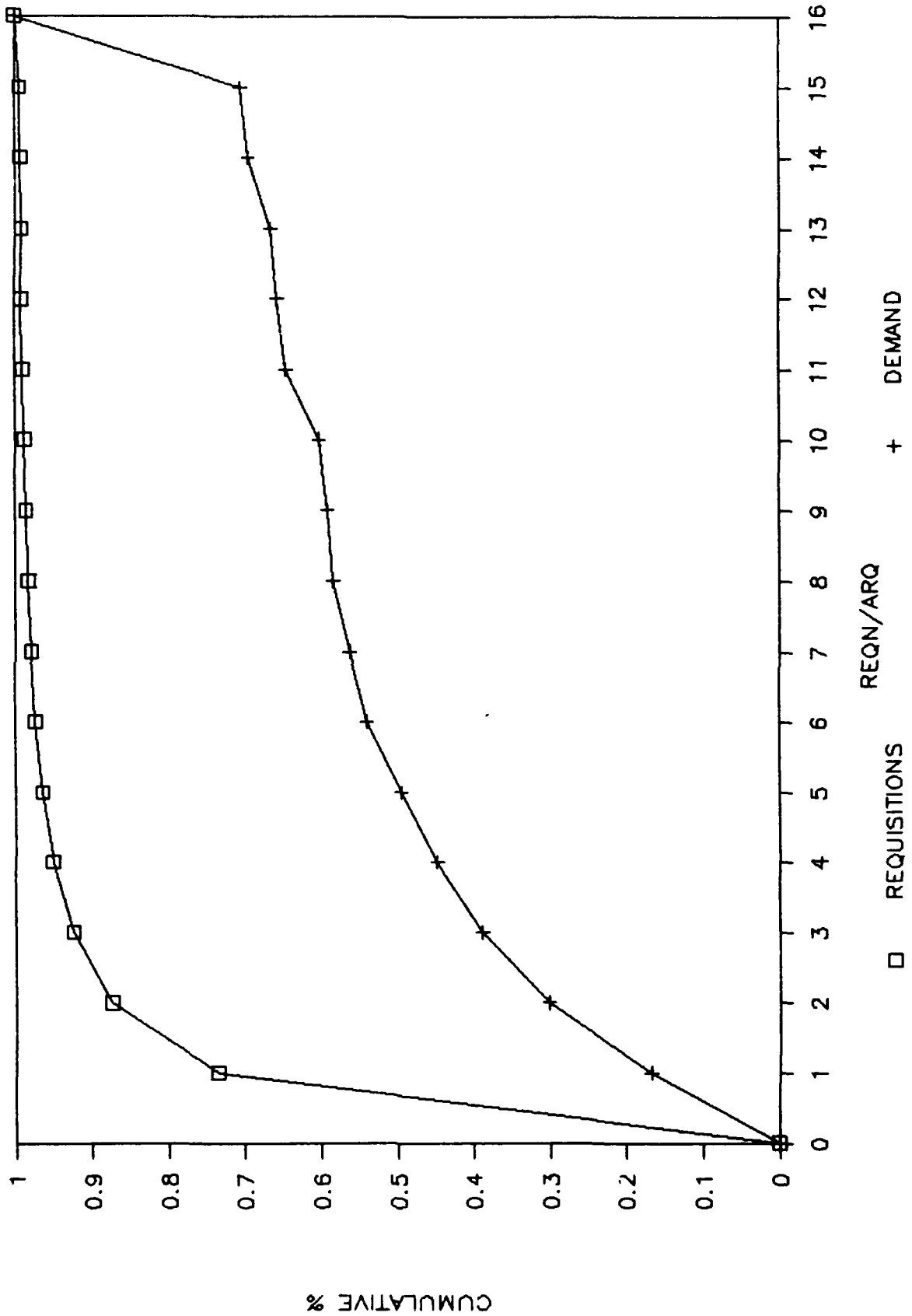
(8) Probability Of A Duein Being Late

The dynamics of the Medical supply system include the fact that scheduled deliveries are frequently late. In order to simulate this portion of the system, a probability distribution was developed using the SAMMS Contract Delinquency Report (USPF0381). This report stratifies the number of contract lines from 31 to 90, 91 to 180, and 181 + days late. In determining how late a contract might be, the delinquency report range (31-90) had to be modified for use in the model. Discussions with Medical personnel indicated that on occasion, a contract might be early. Based on these discussions, it was decided that if an item was going to be early, it was usually 10 days early, and if it was going to be late, 45, 120, or 180 days late were viable point estimates. Using these points, and the information from the Delinquency report, an item has a 7% chance of being 10 days early; a 59% chance of being on time; a 12% chance of

FIGURE 1

BASIC ARQ CONCEPT

MEDICAL DATA FY88



being 45 days late; a 10% chance of being 120 days late; and a 12% chance of being 180 days late. See figure 2.

(9) Other System Parameters

The major system parameter utilized by the model is the requirements objective. At the beginning of the simulation, this figure is composed of the war reserve requirements, special program requirements, the safety level, an item's procurement cycle quantity, and an item's procurement cycle period (expressed as a quantity). During the simulation this figure changes as a result of two factors. The first is the addition of backorders (expressed as a quantity), and the second is the determination of whether to use the procurement cycle quantity or the procurement cycle period (expressed as a quantity) in the calculation. Depending where the simulation is in time, the procurement cycle quantity to be utilized is determined. If the model is making a buy based on the item's procurement lead time, the procurement cycle quantity is utilized. If the model is making a buy on the item's procurement cycle period, the procurement cycle period (expressed as a quantity) is utilized.

(10) Assumptions

(A) Simulated Inventory Policy

The simulation model attempts to emulate exactly what an item manager tries to accomplish, namely, to achieve a given basic inventory level which is a combination of a safety level, operating level, war reserve level, and special program requirements level. The quantity to be procured is obtained by forecasting customer demands and inventory depletion, accounting for existing backorders, and then subtracting inventory on order. Problems in variable leadtimes, changing demand patterns, and system dynamics create unanticipated stock levels complicating item management.

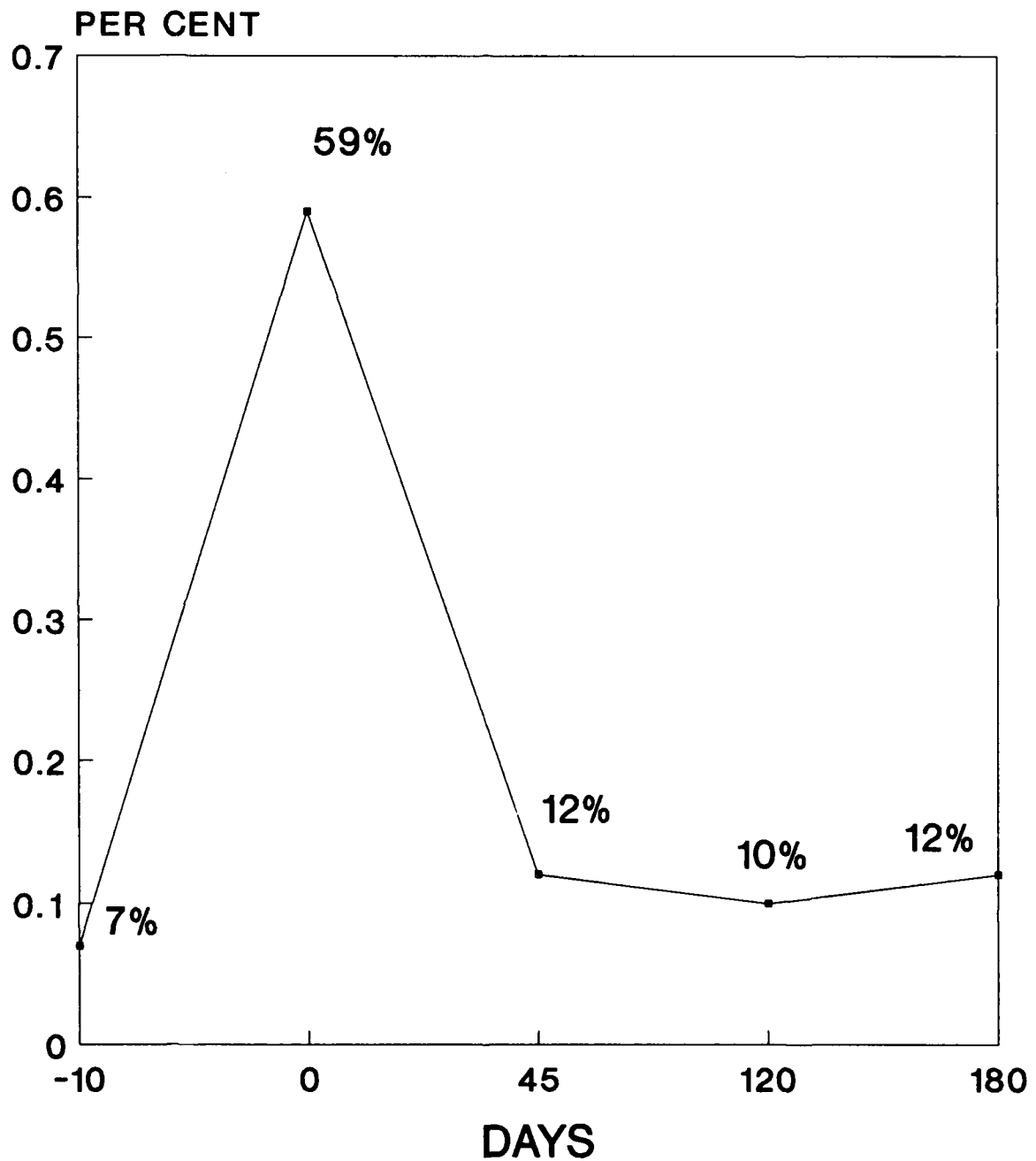
Item managers make inventory replenishment and demand forecasting decisions and initiate requests for the purchase of an item depending on the current requirements objective (See paragraph II.A.(3) above).

(B) Demand Forecasting

The initial forecast utilized by the model is the current system forecast (See paragraph II.C.(7), File 1, item 5 above) provided from the data extraction of the SAMMS files. Once the model begins to simulate the SAMMS process, a new forecast is computed (monthly for the VIP items, and quarterly for all other items) by utilizing the demands (requisition quantities) created during the month or quarter. This is done to create a dynamic inventory process in which the forecasts are changed due to fluctuating demand patterns. The calculations for the double exponential smoothing figure are the same as in paragraph II.A.(5) above, with the single being changed from:

$$S_t = \alpha(P_t - S_{t-1}) + S_{t-1}$$

FIGURE 2
PROBABILITY OF A DUEIN BEING LATE



Duein Ranges from 10 Days Early to 180 Days Late

to the following:

$$S_t = \alpha(TREC - S_{t-1}) + S_{t-1}$$

where:

TREC: The demands (quantity) generated last month or last quarter.

The system expected demand, i.e., the new forecasted demand is then calculated by using the formula presented in paragraph II.B.(3) above.

(C) Generation/Filling of Back Orders

(1) Current SAMMS Process

The model generates and fills backorders the same way the current SAMMS process does. A requisition is generated and then a series of checks takes place prior to either filling the requisition, or putting it on backorder. If the current on hand levels are above the UMMIPS levels, (See paragraph II.A.(1)) the size of the requisition is checked against that item's maximum release quantity. If on hand levels are below the UMMIPS levels, a backorder is generated. If the requisition is less than the maximum release quantity, it is filled and another requisition generated. If the requisition is greater than the maximum release quantity, a partial fill (up to the maximum release quantity) is made, and the remainder put on backorder.

(2) The ARQ Process

When the critical stock position has been breached, the ARQ process checks the requisition's size, and the type (IPG 1,2,3) of requisition generated. The size of the requisition is checked against that item's ARQ. If it is an IPG 1, the requisition is filled. If it is an IPG 2, and the requisition is greater than or equal to three times the ARQ, it is automatically backordered. If it is an IPG 3, the process is identical, only the limit is one and a half times the ARQ. The limits discussed for the IPG 2 and 3 requisitions are only benchmarks, and are altered during the simulation to try and find an optimal mix, i.e., one that reduces the backorders the most.

(3) The Filling of Back Orders

When back orders have been generated, they are included in the calculation of the requirements objective for procurement and subsequent filling. The back orders, in order of precedence and date, are then filled when stock becomes available.

III. ANALYSIS

Utilizing a Monte Carlo simulation to analyze the ARQ process versus the current SAMMS process produced results that looked out five (5) years into the future. The ARQ process was analyzed using the Medical policy that triggers the process when an item reaches a 30

day stock level, that IPG 2 requisitions would be placed on backorder if the requisition size was 3X the ARQ, and an IPG 3 requisition would be placed on backorder if it was 1.5X the ARQ. The results (as measured by the definition in paragraph I.C.) indicated that the ARQ process performed essentially equal in terms of supply availability and backorders generated to the current SAMMS process. This does not mean that the ARQ process does not or will not help supply availability, it only means that in the long run looking over all NSNs as an aggregate, it is neither better nor worse than SAMMS.

An in depth analysis of the results presented above indicate that there are time periods when the ARQ process produces the results desired, and time periods when it does not. In particular, in the latter case as the ARQ process is currently triggered when an item reaches 30 days of stock on hand, if a duein is imminent, large requisitions placed on backorder (because of the ARQ trigger) may be unnecessarily created. This is due to the fact that enough stock is available to cover these requisitions and the UMMIPS levels will not be violated prior to the duein. If the duein is delayed and doesn't come in within 30 days of reaching the critical level, then ARQ will help, but only to the extent that large requisitions happen to come in during this time period.

IV. CONCLUSIONS

Based upon the above, the ARQ inventory policy can probably be compared to an insurance policy. The concept of preserving certain amounts of stock to satisfy requisitions of a lesser priority (not quantities requested) seems to work in those instances when dueins are severely delayed. However, in those instances where dueins are not delayed, the ARQ policy will create unnecessary backorders. Overall, in the long run, it is essentially the same as SAMMS.

The various Federal Supply Classes (FSC's) managed by Medical were also examined. With only cursory data available, it appears that certain of the classes respond better to the ARQ concept while for others, it is detrimental. Two examples are the drug & biological (6505) and optical supply (6540) classes. The drug & biological class seemed to respond positively to the process, while the optical supply class did not. A definitive answer as to why this occurred is unknown, and further study of these classes of supply, along with their requisitioning histories and procurement policies may provide an answer.

V. RECOMMENDATIONS

The following recommendations are made as a result of this study:

A. That the Medical Supply Operations Divisions continue utilizing the ARQ process as it is currently operating, and monitor their supply availability and backorders generated for possible signs that the process is not working as desired. This recommendation is made with the knowledge that Procurement has been able to decrease their backlog of work and provide a much smoother flow of contract awards. If this

continues, then the scenario outlined in the Analysis needs to be monitored much more closely.

B. That the Medical Directorate's MSO review the NIR and SCF SAMMS files for possible updates and corrections in relation to individual items average requisition quantity, item category code, assignment of an NSN and nomenclature, quarterly forecasted demand (especially if it is zero (0), the alpha factor, and lead times, both administrative and procurement.

C. That the Medical Directorate consider an operations research follow-on study to analyze the ARQ process under a myriad of scenarios, i.e., when should the process be triggered, how large should IPG 2 and 3 requisitions be before they are automatically backordered, and should the process be selectively used for various of the FSC's managed by Medical. By simulating a number of different scenarios, and analyzing their effects, it may be possible to determine an optimal mix of items to use under the ARQ process, and when it works the best.

VI. BENEFITS

Quantifying the benefits of implementing an inventory policy that in the long run, does not out perform SAMMS is difficult. ARQ acts as an insurance policy that does help when dueins or procurements are delayed. ARQ however may actually decrease supply availability in those instances when dueins are imminent. If the ARQ policy can be modified to account for the timeliness of expected current dueins, then the downside of an ARQ policy will be eliminated.

Appendix A

Documentation for the Variables Used in the Medical Simulation Model

DOCUMENTATION FOR THE VARIABLES USED IN THE SIMULATION MODEL

The following definitions are provided as documentation for the variables utilized in the simulation model of the Supply Operations System in the Medical Directorate.

<u>VARIABLE</u>	<u>DEFINITION</u>
1. XSIM (See item 50)	The number of passes (days) that the model collects data on each item simulated (1500). Calculated by $(60 \times 30) - 300$. Seed is first 300 days.
2. NSX	The number of simulation passes made by the model (60).
3. NF1	Array utilized to store stock on hand, and replenishment stock during the simulation.
4. YR, YS, JL	Variables utilized in system to open data files used by the simulation, or created by the simulation as output.
5. IDUM, ISUM	Variables used in the random generator subroutine. Idum calls the routine itime and isum is used to calculate the random variable.

The following twenty-eight variables contain the data on each NSN to be analyzed by the model. This data comes from the data provided by OTIS that has been filtered for use by the model.

6. NSN1	The item's NSN.
7. VIP1	Very Important Program item. Needed for forecasting purposes. These items are forecast monthly vs quarterly. Looks for a Y or M in the data field.
10. QFD1	The item's quarterly forecasted demand. Recalculated each quarter or month during the simulation.
11. SYSS1	An initial value for the item's single exponential quantity. Recalculated as necessary during the simulation.
12. SYSD1	An initial value for the item's double exponential quantity. Recalculated as necessary during the simulation.
13. SL1	The item's safety level factor in days.

Page 2 Documentation for the Variables Utilized in the Medical
Simulation Model

- | | | |
|-----|--------|---|
| 14. | ALPHA1 | The item's Alpha factor. Used in the exponential smoothing technique. It's either a .05 or .10. |
| 15. | LT1 | The item's lead time in days. |
| 16. | PC1 | The item's procurement cycle lead time, (expressed as a quantity) in months |
| 17. | SHLF1 | The item's shelf life in months. |
| 18. | P | The item's unit price. |
| 19. | MRQ1 | The maximum release quantity (MRQ) for each individual item (NSN) evaluated. |
| 20. | ICC1 | The item's item category code. |
| 21. | ROP1 | The item's reorder point level.
from Table 018. |
| 22. | OHA1 | The item's on-hand assets at the beginning of the simulation. |
| 23. | SPR1 | The item's special program requirements level--usually DEPMEDS, used in the requirements objective calculation. |
| 24. | WRL1 | The item's war reserve protectable level. Used in the requirements objective calculation. |
| 25. | SUM1 | The number of requisitions an item has received during the past 24 months for those months that it has received them. |
| 26. | ARQ1 | The item's average requisition quantity over the number of months that it has received requisitions during the last 24. |
| 27. | INUM1 | The number of months that an item has had demands in the last 24. |
| 28. | PCP1 | The item's procurement cycle period expressed as a quantity. Used in the requirements objective calculation. |
| 29. | PC2 | The item's procurement cycle quantity--includes admin and plt. Used in the requirements objective calculation. |

Page 3 Documentation for the Variables Utilized in the Medical
Simulation Model

- | | | | |
|-----|-----------------------------|----------------------|--|
| 30. | PCM1 | (See item 35) | The item's procurement cycle period expressed in months. Used to determine which variables are used in the calculation of the requirements obj. |
| 31. | LF | | Variable used in print loop which defines how many NSN's are being evaluated. |
| 32. | KALT | | A trigger used for printing purposes. If it's a 0, no item print is made, if it is a 1, an entire print is made for each NSN. |
| 33. | TFACTOR | | Utilized in the EOQ calculation. Provided by HQ DLA, set at 95 and comes from Policy Table 018. |
| 34. | REQS1 | (See item 48) | The number of requisitions per day that an item receives. |
| 35. | REQOBJ | (See item 30) | An item's requirement objective. Changes during the simulation because the requirements are different at different times. |
| 36. | REP | | The amount of replenishment stock due-in. |
| 37. | IPG1
IPG2
IPG3 | (See item 57) | The counters for the number of IPG 1,2,3 requisitions generated. |
| 38. | IB
IB1
IB2
IB3 | (See item's 52 & 61) | The counters for the number of IPG 1,2,3 backorders generated. IB is a cumulative number. |
| 39. | IBB1
IBB2
IBB3 | (See item 54) | The storage locations for the size of each IPG 1,2,3 backorder generated. Once filled, it is zeroed out. |
| 40. | BO1 | | The same as item 39. Used in the subroutine STDUE to determine if stock needs to be ordered for a backorder not ordered yet. Once ordered, it is zeroed out. |
| 41. | IBZ
IBZ1
IBZ2
IBZ3 | (See item 53) | The counters to determine how many of the IBB1,2,3 backorders were actually filled. IBZ is a cumulative number. |

Page 4 Documentation for the Variables Utilized in the Medical
Simulation Model

- | | | | |
|-----|--------------|--------------------------------|--|
| 42. | IK | (Only used
if OHA <
ROP) | The time period when stock ordered will arrive. Is a combination of an item's actual leadtime.(in days), and the days that the item might be late. The probability of being late is built on the CLIN delinquency report. |
| 43. | TOTDUE | | The amount of stock due in. |
| 44. | TSTOCK | | The amount of stock available for issue. |
| 45. | RM | | The number of requisitions per month that an item receives. |
| 46. | RD | | The number of requisitions per day that an item receives (RM/30). |
| 47. | RZ & LM | | Variables used to tell the model how many days must pass before an item that gets less than one requisition per day should get a requisition. |
| 48. | IRPASS | (See items 34 & 47) | Variable used to determine if enough passes of the simulation have been made for an item which gets less than one (1) requisition/day to get a requisition. |
| 49. | PREDEM | | Used in Custdem subroutine as the previous months demand by the exponential smoothing technique. |
| 50. | IT | (See item 1) | A counter used by the model to determine which pass the simulation is on. The model uses 30 days/month, and it makes 60 passes, making IT run from 1 to 1800. |
| 51. | BL | | Variable that represents the quantity of backorders generated that have not been filled that is checked against the current requirements objective to determine if more stock needs to be ordered. When a backorder is filled, BL is decreased by that amount. |
| 52. | IB2A
IB3A | (See item 38) | A counter that represents the backorder generated in the ARQ process for an IPG 2 (2A) or IPG 3 (3A) requisition. |

Page 5 Documentation for the Variables Utilized in the Medical
Simulation Model

53. IB2A1
IB3A1
(See item 41) A counter that represents a backorder generated by the ARQ process that has been filled. An IPG 2 is 2A1 and an IPG 3 is 3A1.
54. IBB2A
IBB3A
(See item 39) The storage locations for the size of IPG 2/3 backorders generated by the ARQ process.
55. TREC The cumulative quantity of requisitions generated that is utilized in the double exponential smoothing technique. A monthly figure for VIP items, otherwise quarterly.
56. ARQCHK The variable that holds the stock value for each NSN evaluated that is used to determine when the ARQ process should be started. Value is in days of stock.

The following variables are utilized in the print sequence of the model.

57. IZ
(See item 37) A cumulative counter for the number of IPG 1,2,3 requisitions generated during the simulation by NSN evaluated.
58. BB The percent of time that an individual NSN is NIS.
59. ISUM1
ISUM2
ISUM3
IGTOT The counters for the number of IPG 1,2,3 requisitions generated during the simulation. IGTOT is the total number of requisitions generated.
60. P1
P2
P3 The percentage of IPG 1,2,3 requisitions generated during the simulation. Approximates the actual percentages received in Medical.
61. JA (See item 38)
JA1
JA2
JA3 The counters for the cumulative number of IPG 1,2,3 backorders generated during the simulation. JA is the cumulative total.
62. WA The weighted average NIS of all NSN's evaluated.
- WA1 The revised weighted average NIS for all NSN's after backorders filled have been taken into account.

Page 6 Documentation for the Variables Utilized in the Medical
Simulation Model

- | | | |
|-----|-------------------------|--|
| 63. | LA
LA1
LA2
LA3 | The number of IPG 1,2,3 backorders filled. LA is the cumulative number of backorders filled. |
| 64. | IWA2 | The cumulative number of backorders that did not get filled (JA - LA). |
| 65. | IWA3
IWA4
IWA5 | The number of IPG 1,2,3 backorders that did not get filled. A3 = IPG 1, etc. |
| 66. | JA2A
JA3A | The number of IPG 2,3 requisitions generated by the ARQ process. |
| 67. | JA2A1
JA3A1 | The number of IPG 2,3 requisitions generated by the ARQ process that were filled. |

Appendix B

Computer Listing of the SAMMS Simulation
Portion of the Model

```

C Program that runs a simulation model for the Medical
C Directorate. It will be used to test whether or not
C the Average Requisition Quantity (ARQ) concept of
C issuing stock should be continued. The ARQ concept states
C that if an IPG 2 requisition is 3X the ARQ, it goes on
C backorder, and if an IPG 3 requisition is 1.5X the ARQ, it
C goes on backorder. IPG 1 requisitions are always filled.
C The concept will be tested against the current policy, and
C variations in the multiple of the ARQ will also be tested.
C
C To run the program, type: works <medical >junk &
C This runs the executable module sworks, and uses the
C file medical to answer questions asked by the model (in this
C case, how many NSNs will be analyzed), and then prints the
C responses to the questions to the file junk which can be later
C erased. The ampersand (&) runs the program in the background.
C
C The name of the program is sambkor.f It's output file is works
C
C ---THIS VERSION RUNS THE SAMMS VERSION OF THE MODEL---
C ---AND IS BASED ON A 30 DAY MONTH
C
C ---THIS VERSION INCLUDES THE ORDERING OF BACKORDERS
C THAT HAVE BEEN CREATED DURING THE SIMULATION.
C
CHARACTER*1 ICC, VIP, NSN*13, NOM*30, NOM1(950)*30
CHARACTER*1 ICC1(950), VIP1(950), NSN1(950)*13
INTEGER YR, YS, TFACTOR, PCM, PCM1(950), ALT, PLT
INTEGER CTR, LT1(950), PC1(950), SHLF1(950), MONTHS, PC, SHLF
REAL INUM1(950), IBB1(60,30,50), IBB2(60,30,50)
REAL IBB3(60,30,50)
DIMENSION MRQ1(950), ARQ1(950), SPR1(950), IB(950), IZ(950)
DIMENSION IPG1(950), IPG2(950), IPG3(950), SUM1(950), FORDEM(950)
DIMENSION IBZ(950), IBZ1(950), IBZ2(950), IBZ3(950)
DIMENSION STOH(9100), REP(9100), IL(50), PREDEM(950)
DIMENSION IB1(950), IB2(950), IB3(950), PCP1(950), IREQS1(950)
DIMENSION TSTOCK(1850), REQOBJ(62), BO1(60,30,50), BO2(60,30,50)
DIMENSION IR1A(50), IR2A(50), IR3A(50), BO3(60,30,50), SL1(950)
DIMENSION QFD1(950,62), OHA1(950), ROP1(950), WRL1(950), P(950)
DIMENSION SYSS1(950,62), SYSD1(950,62), MAD1(950), ALPHA1(950)
DIMENSION SING(950,62), DOUB(950,62), ARQCHK(950), PC2(950)
COMMON IDUM, ISUM
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1, P, SL1, PC, J, R,
*MONTHS, SDUEIN, PC2, SHLF1, DUEIN, IIT, LT1
COMMON /BLK5/ISIM, SYSS1, SYSD1, MAD1, IT
COMMON /BLK6/ALPHA1, SING, DOUB
COMMON /BLK2/VIP1
COMMON /BLK3/ARQCHK, JK, TREC, PREDEM, FORDEM
COMMON /BLK4/BO1, BO2, BO3, M1L, M2L, M3L, IR1A, IR2A, IR3A
IDUM = 100
R = RAND(IDUM)
WRITE(6, '("How many items will the model be using ?")')
READ(5,11) N
11 FORMAT(I5)

```



```

        WRITE(6,12) N
12 FORMAT(/,'The model is using',I5,' items')
C
C NSX is the number of simulation passes through the program
C
        XSIM = 1100
        NSX = 60
        NF1 = NSX * 150
C
C The following is the set variable for the EOQ calculation
C
        TFACTOR = 95
C
C The data files are now opened for use
C These files contain the data used in calculations in the
C simulation.
C
        YR = 33
        YS = 34
        OPEN(YR,FILE='med1a')
        OPEN(YS,FILE='med2a')
        REWIND 33
        REWIND 34
        READ(YR,15)
        READ(YS,15)
15 FORMAT(/,/,,/,/,/,/,/,/)
        JL = 53
        OPEN(JL,FILE='response3')
C
C This is the loading of the data for the simulation and the
C beginning of the simulation itself.
C
C The variables are defined as follows:
C CTR--A counter used to identify each NSN
C NSN--The items NSN
C UP--The items unit price
C VIP--A key to determine if the item is a VIP item or not
C DV--The demand value code of an item
C ICC--The item's item category code
C MRQ--The item's maximum release quantity
C OHA--The on hand assets of the item at the beginning of the study
C ROP--The reorder point level of the item
C ARQ--The item's average requisition quantity
C QFD--The item's quarterly forecasted demand (It's monthly if VIP)
C SYSS--The item's system single exponential quantity
C SYSD--The item's system double exponential quantity
C SL--The item's safety level in days
C ALPHA--The item's alpha factor for exponential smoothing
C LT--The item's lead time in days
C PCM--The item's procurement lead time in months
C SHLF--The item's shelf life in months
C WRL--The item's war reserve protectable level
C SPR--The item's special program requirements-usually DEPMEDS
C MAD--The item's mean absolute deviation

```

C EEX--The item's historical average quarterly demand
 C ESTD--The item's historical standard deviation of its EEX
 C PCQ--The item's procurement cycle quantity--includes admin and plt
 C PCPQ--The item's procurement cycle period expressed as a quantity.
 C NOM--The item's nomenclature
 C PC--The item's procurement cycle period expressed in months
 C REQS--The item's number of requisitions per day
 C INUM--The number of months an item has had demands
 C SUM--The number of requisitions received during the months demanded
 C

```

      DO 20 IJ = 1, N
      READ(YR,100,END=50,ERR=900) CTR,NSN,NOM,VIP,QFD,SYSS,SYSD,
      *SL,ALPHA,ALT,PLT,PC,SHLF,MAD
100  FORMAT(1X,I5,1X,A13,1X,A19,2X,A1,10X,F7.0,1X,F8.1,1X,F8.1,
      *F7.0,2X,F5.2,2X,I3,2X,I3,2X,I3,4X,I2,2X,F8.1)
      NSN1(CTR) = NSN
      NOM1(CTR) = NOM
      VIP1(CTR) = VIP
      QFD1(CTR,1) = QFD
      SYSS1(CTR,1) = SYSS
      SYSD1(CTR,1) = SYSD
      SL1(CTR) = SL
      ALPHA1(CTR) = ALPHA
      LT1(CTR) = ALT + PLT
      PC1(CTR) = PC
      SHLF1(CTR) = SHLF
      MAD1(CTR) = MAD
      READ(YS,101,END=51,ERR=900) CTR,UP,MRQ,ICC,ROP,OHA,SPR,WRL,
      *SUM,ATOT,INUM,PCPQ,PCQ,PCM
101  FORMAT(1X,I5,14X,F8.2,1X,F6.0,4X,A1,F8.0,1X,F9.0,F8.0,F9.0,
      *4X,F6.0,3X,F8.2,4X,I3,2X,F8.0,1X,F8.0,I8)
      P(CTR) = UP
      MRQ1(CTR) = MRQ
      ICC1(CTR) = ICC
      ROP1(CTR) = ROP
      OHA1(CTR) = OHA
      SPR1(CTR) = SPR
      WRL1(CTR) = WRL
      SUM1(CTR) = SUM
      ARQ1(CTR) = ATOT
      INUM1(CTR) = INUM
      PCP1(CTR) = PCPQ
      PC2(CTR) = PCQ
      PCM1(CTR) = PCM
20  CONTINUE
50  CLOSE (YR)
51  CLOSE (YS)
      LF = N
  
```

C
 WRITE(JL,699)
 699 FORMAT(50X,'CURRENT SAMMS INVENTORY PROCEDURE',//)
 WRITE(JL,700)
 700 FORMAT(38X,'# REQS',4X,'AVE')
 WRITE(JL,701)

```

701 FORMAT(36X,'DURING LAST',1X,'REQ',3X,'MONTHS',6X,'NUMBER OF REQUIS
*ITIONS',6X,'% TIMES',1X,'CUMULATIVE',6X,'BACKORDERS BY')
WRITE(JL,702)
702 FORMAT(4X,'NSN',10X,'NOMENCLATURE',7X,'24 MONTHS',3X,'SIZE',1X,'DE
*MANDED',3X,'IPG1',4X,'IPG2',4X,'IPG3',4X,'TOTAL',3X,
*'NIS',3X,'BACKORDERS',4X,'IPG1',2X,'IPG2',2X,'IPG3',//)

```

C
C Trigger to print full output for an analysis of a particular
C NSN or group of NSNs. If KALT = 0, no print out is made, if
C KALT = 1, the full print out is made.

C
C KALT = 0

C
C Actual start of the simulation

C
C
C JJ = 43
C OPEN(JJ,FILE='CHECK3')
C DO 210 J = 1, N
C REQOBJ(J) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J) + PCP1(J)
C REQXXX = REQOBJ(J)
C IF(OHA1(J) .GT. REQOBJ(J)) THEN
C GO TO 55
C ELSE
C OHA1(J) = REQOBJ(J)
C ENDIF
C 55 TOTDUE = 0.
C IK = 0
C MA = 1
C IL(MA) = 0
C WRITE(JJ,650) NSN1(J), REQOBJ(J)
C 650 FORMAT('The model is processing NSN ',A13,' with requirements obj
C *=',F10.2)
C JK = 4
C UMMIPSA = SL1(J) * .8
C IF(VIP1(J) .EQ. 'Y' .OR. VIP1(J) .EQ. 'M') THEN
C UMMIPSB = QFD1(J,1)/2.
C ELSE
C UMMIPSB = QFD1(J,1)/6.
C ENDIF
C IF (UMMIPSA .LT. UMMIPSB) THEN
C UMMIPS = UMMIPSA
C ELSE
C UMMIPS = UMMIPSB
C ENDIF
C DO 120 LZ = 1, NF1
C STOH(LZ) = 0.
C REP(LZ) = 0.
C 120 CONTINUE
C DO 121 L1 = 1,60
C DO 122 L2 = 1,30
C DO 123 L3 = 1,50
C IBB1(L1,L2,L3) = 0.
C IBB2(L1,L2,L3) = 0.
C IBB3(L1,L2,L3) = 0.

```

      BO1(L1,L2,L3) = 0.
      BO2(L1,L2,L3) = 0.
      BO3(L1,L2,L3) = 0.
123 CONTINUE
122 CONTINUE
121 CONTINUE
      IPG1(J) = 0
      IPG2(J) = 0
      IPG3(J) = 0
      IBZ(J) = 0
      IBZ1(J) = 0
      IBZ2(J) = 0
      IBZ3(J) = 0
      IB(J) = 0
      IB1(J) = 0
      IB2(J) = 0
      IB3(J) = 0
      IRPASS = 1
      LTOT = PC1(J)
      LTOT1 = PCM1(J)
      QFD1(J,1) = QFD1(J,1)
      SYSS1(J,1) = SYSS1(J,1)
      SYSD1(J,1) = SYSD1(J,1)
      IF (ROP1(J) .GT. OHA1(J)) THEN
      CALL STDUEIN
      CALL ARRIVE
      IK = IK + IIT
      TOTDUE = TOTDUE + DUEIN
      REP(IK) = DUEIN
      ELSE
      ENDIF
      IT = 1
      TSTOCK(IT) = OHA1(J)
      IT = 0
      RM = SUM1(J)/INUM1(J)
      RD = RM/30.
      IF(RD .LT. 1.) THEN
      LM = INT((1./RD) + .5)
      RZ = LM
      IRPASS = IRPASS + LM
      PREDEM(J) = (INT((30./RZ) + .5) * ARQ1(J))
      IREQS1(J) = 0
      ELSE
      IREQS1(J) = INT(RD + .5)
      PREDEM(J) = (IREQS1(J) * ARQ1(J)) * 30.
      ENDIF
      BL = 0.
      LB = 0
      TREC = 0.
      FORDEM(J) = 0.
      NUMREQ = IREQS1(J)
      M1L = 1
      M2L = 1
      M3L = 1

```

```

DO 200 ISIM = 1, NSX
IF(VIP1(J) .EQ. 'N' .AND. ISIM .LE. 3) THEN
QFD1(J,ISIM) = QFD1(J,1)
ELSE
ENDIF
LB = LB + 1
IF((ISIM .LT. LTOT) .OR. (ISIM .LT. LTOT1)) THEN
REQOBJ(ISIM) = REQXXX
ELSE
ENDIF
IF(PC1(J) .EQ. PCM1(J)) GO TO 320

```

C
C This following loop takes care of the case when ltot = ltot1 = isim.
C It uses the smaller of pc2 or pcpl to get more stock ordered.
C

```

IF(LTOT .EQ. LTOT1 .AND. LTOT .EQ. ISIM) THEN
IF(PC2(J) .LT. PCP1(J)) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J)
LTOT = LTOT + PC1(J)
LTOT1 = LTOT1 + PCM1(J)
GO TO 321
ELSE
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PCP1(J)
LTOT = LTOT + PC1(J)
LTOT1 = LTOT1 + PCM1(J)
GO TO 321
ENDIF
ELSE
ENDIF

```

C
C End of the loop for the special case
C

```

IF(ISIM .EQ. LTOT) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J)
LTOT = LTOT + PC1(J)
GO TO 321
ELSE
ENDIF
IF (ISIM .EQ. LTOT1) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PCP1(J)
LTOT1 = LTOT1 + PCM1(J)
ELSE
ENDIF
GO TO 321
320 IF (ISIM .EQ. LTOT) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J) + PCP1(J)
LTOT = LTOT + PC1(J)
ELSE
ENDIF
321 IF(VIP1(J) .EQ. 'N' .AND. LB .EQ. 4) THEN
FORDEM(J) = 0.
CALL CUSTDEM
TREC = 0.
LB = 1

```

```

ELSE
ENDIF
IF(VIP1(J) .EQ. 'N' .AND. ISIM .GT. 3) THEN
QFD1(J,ISIM) = FORDEM(J)
ELSE
ENDIF
IF(VIP1(J) .EQ. 'Y' .OR. VIP1(J) .EQ. 'M') THEN
CALL CUSTDEM
TREC = 0.
ELSE
ENDIF
DO 190 I = 1, 30
IT = IT + 1
TSTOCK(IT) = TSTOCK(IT) + REP(IT)
IF(KALT .EQ. 1) THEN
WRITE(JJ,87) IT,TSTOCK(IT),REP(IT),BL,TOTDUE,JI,REP(JI),ISIM,
*REQOBJ(ISIM)
87 FORMAT(2X,I5,4(3X,F10.2),5X,I4,5X,F10.2,2X,I4,2X,F10.2)
ELSE
ENDIF
IF(RD .LT. 1. .AND. IT .GT. 1) THEN
IF(IT .EQ. IRPASS) THEN
IRPASS = IRPASS + LM
GO TO 140
ELSE
ENDIF
ELSE
ENDIF
IF(RD .LT. 1. .AND. IT .EQ. 1) GO TO 130

```

C
C Beginning of the process to fill backorders in order of IPG group
C and time, i.e., when the backorder was generated
C

```

140 IF(TSTOCK(IT) .EQ. 0. .OR. IB(J) .EQ. 0 .OR. IT .LE. 300
*) GO TO 141
IF((IB1(J) - IBZ1(J)) .GT. 0) THEN
IF(RD .GT. 1.) THEN
IL1 = IREQS1(J)
ELSE
IL1 = M1L
ENDIF
DO 441 M1 = 11, ISIM
DO 550 K1 = 1, 30
DO 551 N1 = 1, IL1
IF(IBM1(M1,K1,IR1A(N1)) .EQ. 0.) GO TO 949
IF(TSTOCK(IT) .LT. IBM1(M1,K1,IR1A(N1))) GO TO 141
TSTOCK(IT) = TSTOCK(IT) - IBM1(M1,K1,IR1A(N1))
IF(TSTOCK(IT) .EQ. 0.) THEN
IBZ(J) = IBZ(J) + 1
IBZ1(J) = IBZ1(J) + 1
TSTOCK(IT) = 0.
BL = BL - IBM1(M1,K1,IR1A(N1))
IBM1(M1,K1,IR1A(N1)) = 0.
GO TO 141

```

```

ELSE
  IBZ(J) = IBZ(J) + 1
  IBZ1(J) = IBZ1(J) + 1
  BL = BL - IBB1(M1,K1,IR1A(N1))
  IBB1(M1,K1,IR1A(N1)) = 0.
ENDIF
949 CONTINUE
551 CONTINUE
550 CONTINUE
441 CONTINUE
ELSE
ENDIF
IF(TSTOCK(IT) .LE. 0.) THEN
  TSTOCK(IT) = 0.
  GO TO 141
ELSE
ENDIF
IF((IBZ(J) - IBZ2(J)) .GT. 0) THEN
  IF(RD .GT. 1.) THEN
    IL2 = IREQS1(J)
  ELSE
    IL2 = M2L
  ENDIF
  DO 442 M2 = 11, ISIM
  DO 552 K2 = 1, 30
  DO 553 N2 = 1, IL2
    IF(IBB2(M2,K2,IR2A(N2)) .EQ. 0.) GO TO 959
    IF(TSTOCK(IT) .LT. IBB2(M2,K2,IR2A(N2))) GO TO 141
    TSTOCK(IT) = TSTOCK(IT) - IBB2(M2,K2,IR2A(N2))
    IF(TSTOCK(IT) .EQ. 0.) THEN
      IBZ(J) = IBZ(J) + 1
      IBZ2(J) = IBZ2(J) + 1
      TSTOCK(IT) = 0.
      BL = BL - IBB2(M2,K2,IR2A(N2))
      IBB2(M2,K2,IR2A(N2)) = 0.
      GO TO 141
    ELSE
      IBZ(J) = IBZ(J) + 1
      IBZ2(J) = IBZ2(J) + 1
      BL = BL - IBB2(M2,K2,IR2A(N2))
      IBB2(M2,K2,IR2A(N2)) = 0.
    ENDIF
  959 CONTINUE
  553 CONTINUE
  552 CONTINUE
  442 CONTINUE
ELSE
ENDIF
IF(TSTOCK(IT) .LE. 0.) THEN
  TSTOCK(IT) = 0.
  GO TO 141
ELSE
ENDIF
IF((IB3(J) - IBZ3(J)) .GT. 0) THEN

```

```

IF(RD .GT. 1.) THEN
IL3 = IREQS1(J)
ELSE
IL3 = M3L
ENDIF
DO 443 M3 = 11, ISIM
DO 554 K3 = 1, 30
DO 555 N3 = 1, IL3
IF(IBB3(M3,K3,IR3A(N3)) .EQ. 0.) GO TO 969
IF(TSTOCK(IT) .LT. IBB3(M3,K3,IR3A(N3))) GO TO 141
TSTOCK(IT) = TSTOCK(IT) - IBB3(M3,K3,IR3A(N3))
IF(TSTOCK(IT) .EQ. 0.) THEN
IBZ(J) = IBZ(J) + 1
IBZ3(J) = IBZ3(J) + 1
TSTOCK(IT) = 0.
BL = BL - IBB3(M3,K3,IR3A(N3))
IBB3(M3,K3,IR3A(N3)) = 0.
GO TO 141
ELSE
IBZ(J) = IBZ(J) + 1
IBZ3(J) = IBZ3(J) + 1
BL = BL - IBB3(M3,K3,IR3A(N3))
IBB3(M3,K3,IR3A(N3)) = 0.
ENDIF
969 CONTINUE
555 CONTINUE
554 CONTINUE
443 CONTINUE
ELSE
ENDIF

```

C
C End of the process to fill backorders
C

```

141 IF(RD .GT. 1.) THEN
GO TO 142
ELSE IF(IT .EQ. (IRPASS - LM)) THEN
GO TO 130
ELSE IF(RD .GT. .95 .AND. RD .LT. 1.05) THEN
GO TO 130
ELSE
GO TO 622
ENDIF

```

C
C Section that generates multiple requisitions per day, given
C that an NSN produces more than one requisition per day.
C

```

142 DO 156 L = 1, NUMREQ
IF(L .EQ. 1) THEN
M1L = 1
M2L = 1
M3L = 1
ELSE
ENDIF
CALL REQ1(ARQ1(J), Y, R)

```



```

TREC = TREC + Y
IF(IT .GT. 300) THEN
IA = R * 100
IF (IA .LE. 9) IPG1(J) = IPG1(J) + 1
IF (IA .GT. 9 .AND. IA .LE. 31) IPG2(J) = IPG2(J) + 1
IF (IA .GT. 31) IPG3(J) = IPG3(J) + 1
ELSE
ENDIF
IF (TSTOCK(IT) .EQ. 0. .AND. IT .GT. 300) THEN
IB(J) = IB(J) + 1
IF (IA .LE. 9) THEN
IB1(J) = IB1(J) + 1
IR1A(M1L) = L
IBB1(ISIM,I,IR1A(M1L)) = Y
BO1(ISIM,I,IR1A(M1L)) = Y
BL = BL + IBB1(ISIM,I,IR1A(M1L))
GO TO 125
ELSE
ENDIF
IF (IA .GT. 9 .AND. IA .LE. 31) THEN
IB2(J) = IB2(J) + 1
IR2A(M2L) = L
IBB2(ISIM,I,IR2A(M2L)) = Y
BO2(ISIM,I,IR2A(M2L)) = Y
BL = BL + IBB2(ISIM,I,IR2A(M2L))
GO TO 125
ELSE
ENDIF
IF (IA .GT. 31) THEN
IB3(J) = IB3(J) + 1
IR3A(M3L) = L
IBB3(ISIM,I,IR3A(M3L)) = Y
BO3(ISIM,I,IR3A(M3L)) = Y
BL = BL + IBB3(ISIM,I,IR3A(M3L))
GO TO 125
ELSE
ENDIF
ELSE
ENDIF
IF (Y .GT. MRQ1(J)) THEN
MAXREL = Y - MRQ1(J)
TSTOCK(IT) = TSTOCK(IT) - MAXREL
IF (TSTOCK(IT) .LE. 0.) TSTOCK(IT) = 0.
GO TO 125
ELSE
ENDIF
TSTOCK(IT) = TSTOCK(IT) - Y
IF(TSTOCK(IT) .LE. 0.) TSTOCK(IT) = 0.
125 M1L = M1L + 1
M2L = M2L + 1
M3L = M3L + 1
156 CONTINUE
TSTOCK(IT + 1) = TSTOCK(IT)
IF(TSTOCK(IT + 1) .LE. 0.) TSTOCK(IT + 1) = 0.

```

```

C
C End of the multiple requisition process
C
      GO TO 126
C
C Process to generate a single requisition per day, if only one per
C NSN is required
C
130 CALL REQN(ARQ1(J), Y, R)
      TREC = TREC + Y
      IF(IT .GT. 300) THEN
        IA = R * 100
        IF (IA .LE. 9) IPG1(J) = IPG1(J) + 1
        IF (IA .GT. 9 .AND. IA .LE. 31) IPG2(J) = IPG2(J) + 1
        IF (IA .GT. 31) IPG3(J) = IPG3(J) + 1
      ELSE
      ENDIF
      IF (TSTOCK(IT) .EQ. 0. .AND. IT .GT. 300) THEN
        IB(J) = IB(J) + 1
        IF (IA .LE. 9) THEN
          IB1(J) = IB1(J) + 1
          IR1A(M1L) = 1
          IBB1(ISIM,I,IR1A(M1L)) = Y
          BO1(ISIM,I,IR1A(M1L)) = Y
          BL = BL + IBB1(ISIM,I,IR1A(M1L))
          GO TO 622
        ELSE
        ENDIF
        IF (IA .GT. 9 .AND. IA .LE. 31) THEN
          IB2(J) = IB2(J) + 1
          IR2A(M2L) = 1
          IBB2(ISIM,I,IR2A(M2L)) = Y
          BO2(ISIM,I,IR2A(M2L)) = Y
          BL = BL + IBB2(ISIM,I,IR2A(M2L))
          GO TO 622
        ELSE
        ENDIF
        IF (IA .GT. 31) THEN
          IB3(J) = IB3(J) + 1
          IR3A(M3L) = 1
          IBB3(ISIM,I,IR3A(M3L)) = Y
          BO3(ISIM,I,IR3A(M3L)) = Y
          BL = BL + IBB3(ISIM,I,IR3A(M3L))
          GO TO 622
        ELSE
        ENDIF
      ELSE
      ENDIF
      IF (Y .GT. MRQ1(J)) THEN
        MAXREL = Y - MRQ1(J)
        TSTOCK(IT + 1) = TSTOCK(IT) - MAXREL
        IF (TSTOCK(IT + 1) .LE. 0.) TSTOCK(IT + 1) = 0.
        GO TO 126
      ELSE

```

```

ENDIF
TSTOCK(IT + 1) = TSTOCK(IT) - Y
IF(TSTOCK(IT + 1) .LE. 0.) TSTOCK(IT + 1) = 0.
GO TO 126
622 TSTOCK(IT + 1) = TSTOCK(IT)
C
C End of the requisition generating process
C
126 IF(IT .EQ. IK) THEN
TOTDUE = TOTDUE - REP(IT)
ELSE
ENDIF
DO 250 KJ = 1,MA
IF(IT .EQ. IL(KJ)) THEN
TOTDUE = TOTDUE - REP(IL(KJ))
GO TO 170
ELSE
ENDIF
250 CONTINUE
170 CONTINUE
C
C Section that orders stock, if necessary
C
IF((TSTOCK(IT + 1) + TOTDUE - BL) .LE. REQOBJ(ISIM)) THEN
CALL STDUE(RD,IREQS1(J))
CALL ARRIVE
JI = IT + IIT
IF(IL(MA) .EQ. JI) THEN
REP(JI) = REP(JI) + SDUEIN
GO TO 181
ELSE
ENDIF
MA = MA + 1
IL(MA) = JI
REP(JI) = SDUEIN
181 TOTDUE = TOTDUE + SDUEIN
ELSE
ENDIF
190 CONTINUE
200 CONTINUE
210 CONTINUE
JA = 0
JA1 = 0
JA2 = 0
JA3 = 0
ISUM1 = 0
ISUM2 = 0
ISUM3 = 0
IGTOT = 0
LA = 0
LA1 = 0
LA2 = 0
LA3 = 0
DO 303 K = 1, LF

```

```

JA = JA + IB(K)
JA1 = JA1 + IB1(K)
JA2 = JA2 + IB2(K)
JA3 = JA3 + IB3(K)
ISUM1 = ISUM1 + IPG1(K)
ISUM2 = ISUM2 + IPG2(K)
ISUM3 = ISUM3 + IPG3(K)
LA = LA + IBZ(K)
LA1 = LA1 + IBZ1(K)
LA2 = LA2 + IBZ2(K)
LA3 = LA3 + IBZ3(K)
IZ(K) = IPG1(K) + IPG2(K) + IPG3(K)
AB = IZ(K)
BB = (IB(K)/AB)*100.
WRITE(JL,504)NSN1(K),NOM1(K),SUM1(K),ARQ1(K),INUM1(K),IPG1(K),IP
*G2(K),IPG3(K),IZ(K),BB,IB(K),IB1(K),IB2(K),IB3(K)
504 FORMAT(A13,2X,A22,F6.0,1X,F8.2,3X,F3.0,5X,I4,4X,I4,4X,I5,3X,I6,
*2X,F6.2,1X,I6,6X,I5,1X,I5,2X,I5)
303 CONTINUE
IGTOT = IGTOT + ISUM1 + ISUM2 + ISUM3
XSUM1 = ISUM1
XSUM2 = ISUM2
XSUM3 = ISUM3
XIGTOT = IGTOT
P1 = (XSUM1/XIGTOT)*100.
P2 = (XSUM2/XIGTOT)*100.
P3 = (XSUM3/XIGTOT)*100.
WA = (JA/XIGTOT)*100.
WA1 = ((JA - LA)/XIGTOT)*100.
IWA2 = JA - LA
IWA3 = JA1 - LA1
IWA4 = JA2 - LA2
IWA5 = JA3 - LA3
WRITE(JL,508)ISUM1,ISUM2,ISUM3,IGTOT,JA,JA1,JA2,JA3,P1,P2,P3
508 FORMAT(/,54X,'TOTALS',1X,I6,2X,I6,3X,I6,3X,I6,9X,I6,6X,I5,1X,I
*5,2X,I5,/,50X,'% OF TOTAL',2X,F5.2,3X,F5.2,4X,F5.2)
WRITE(JL,509) WA
509 FORMAT(/,74X,'WEIGHTED AVG NIS',6X,F5.2)
WRITE(JL,528) LA, LA1, LA2, LA3
528 FORMAT(/,73X,'BACKORDERS FILLED',12X,I6,6X,I5,1X,I5,2X,I5)
WRITE(JL,529) WA1, IWA2, IWA3, IWA4, IWA5
529 FORMAT(/,52X,'REVISED NIS (AFTER FILLING BACKORDERS)',6X,F5.2,1X,
*I6,6X,I5,1X,I5,2X,I5)
STOP
900 WRITE(6,('ERROR ENCOUNTERED'))
END

```

```

C
C Subroutine Stduein
C Is the seed for the model. It tests if the current on-hand assets
C are at a reorder point (if so it generates an order), or if they
C are sufficient to begin the simulation. If an order is generated,
C the model uses the SAMMS criteria outlined in the EOQ process.
C

```

```

SUBROUTINE STDUEIN
INTEGER SHLF1(950), MONTHS, TFACTOR
DIMENSION QFD1(950,62), OHA1(950), ROP1(950), P(950), PC2(950)
DIMENSION WRL1(950), SL1(950), SHLF1(950), LT1(950)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1, P, SL1, PC, J, R,
*MONTHS, SDUEIN, PC2, SHLF1, DUEIN, IIT, LT1
DVQD = QFD1(J,1) * P(J)
IF (DVQD .LE. 62. .AND. SHLF1(J) .GE. 36) THEN
DUEIN = 12. * QFD1(J,1)
ELSE IF (DVQD .LE. 62. .AND. SHLF1(J) .LT. 36) THEN
IF (SHLF1(J) .EQ. 0) THEN
DUEIN = PC2(J)
ELSE
DUEIN = (QFD1(J,1)/3.) * SHLF1(J)
ENDIF
ELSE IF (DVQD .GT. 62. .AND. DVQD .LE. 1846.) THEN
EOQ = TFACTOR * (SQRT(DVQD))
MONTHS = (EOQ/P(J))/(QFD1(J,1)/3.)
IF (SHLF1(J) .LT. MONTHS) THEN
IF (SHLF1(J) .EQ. 0) THEN
DUEIN = PC2(J)
ELSE
XX = SHLF1(J)
XXN = MONTHS
DUEIN = (EOQ/P(J)) * (XX/XXN)
ENDIF
ELSE
DUEIN = EOQ/P(J)
ENDIF
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.GE. 9) THEN
DUEIN = 3. * QFD1(J,1)
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.LT. 9) THEN
IF (SHLF1(J) .EQ. 0) THEN
DUEIN = PC2(J)
ELSE
DUEIN = (QFD1(J,1)/3.) * SHLF1(J)
ENDIF
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .GE. 6) THEN
DUEIN = 2. * QFD1(J,1)
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .LT. 6) THEN
IF (SHLF1(J) .EQ. 0) THEN
DUEIN = PC2(J)
ELSE
DUEIN = (QFD1(J,1)/3.) * SHLF1(J)
ENDIF
ENDIF
RETURN
END

```

C

C Subroutine Stdue

C Is used in the simulation to test if stock has fallen below its
C reorder point (if so it generates an order). If an order is

C generated, the simulation uses the SAMMS criteria outlined in
C the EOQ process, and includes backorders not yet accounted for,
C i.e., ordered yet.
C

```

SUBROUTINE STDUE(XRD,KIR)
INTEGER SHLF1(950), MONTHS, TFACTOR
DIMENSION QFD1(950,62),OHA1(950),ROP1(950),WRL1(950),SL1(950)
DIMENSION SYSS1(950,62),SYSD1(950,62),MAD1(950),P(950)
DIMENSION IR1A(50),IR2A(50),IR3A(50), LT1(950)
DIMENSION PC2(950),BO1(60,30,50),BO2(60,30,50),BO3(60,30,50)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1,P,SL1,PC,J,R,
*MONTHS, SDUEIN,PC2,SHLF1,DUEIN,IIT, LT1
COMMON /BLK5/ISIM, SYSS1, SYSD1, MAD1, IT
COMMON /BLK4/BO1,BO2, BO3,M1L,M2L,M3L,IR1A,IR2A,IR3A
BO = 0.
IF(IT .GT. 300) THEN
IF(XRD .GT. 1.) THEN
KL1 = KIR
KL2 = KIR
KL3 = KIR
ELSE
KL1 = M1L
KL2 = M2L
KL3 = M3L
ENDIF
DO 25 KZ = 11, ISIM
DO 30 KA = 1,30
DO 35 LP = 1, KL1
IF(IR1A(LP) .EQ. 0) GO TO 35
BO = BO + BO1(KZ,KA,IR1A(LP))
BO1(KZ,KA,IR1A(LP)) = 0.
35 CONTINUE
30 CONTINUE
25 CONTINUE
DO 26 JE = 11, ISIM
DO 31 KE = 1,30
DO 36 LE = 1, KL2
BO = BO + BO2(JE,KE,IR2A(LE))
BO2(JE,KE,IR2A(LE)) = 0.
36 CONTINUE
31 CONTINUE
26 CONTINUE
DO 27 JD = 11, ISIM
DO 32 KD = 1,30
DO 37 LD = 1, KL3
BO = BO + BO3(JD,KD,IR3A(LD))
BO3(JD,KD,IR3A(LD)) = 0.
37 CONTINUE
32 CONTINUE
27 CONTINUE
ELSE
ENDIF
DVQD = QFD1(J,ISIM) * P(J)
IF (DVQD .LE. 62. .AND. SHLF1(J) .GE. 36) THEN

```

```

SDUEIN = (12. * QFD1(J,ISIM)) + BO
ELSE IF(DVQD .LE. 62. .AND. SHLF1(J) .LT. 36) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
SDUEIN = (QFD1(J,ISIM)/3.) * SHLF1(J) + BO
ENDIF
ELSE IF (DVQD .GT. 62. .AND. DVQD .LE. 1846.) THEN
EOQ = TFACTOR * (SQRT(DVQD))
MONTHS = (EOQ/P(J))/(QFD1(J,ISIM)/3.)
IF (SHLF1(J) .LT. MONTHS) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
XX = SHLF1(J)
XXN = MONTHS
SDUEIN = ((EOQ/P(J)) * (XX/XXN)) + BO
ENDIF
ELSE
SDUEIN = EOQ/P(J) + BO
ENDIF
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.GE. 9) THEN
SDUEIN = (3. * QFD1(J,ISIM)) + BO
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.LT. 9) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
SDUEIN = ((QFD1(J,ISIM)/3.) * SHLF1(J)) + BO
ENDIF
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .GE. 6) THEN
SDUEIN = (2. * QFD1(J,ISIM)) + BO
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .LT. 6) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
SDUEIN = ((QFD1(J,ISIM)/3.) * SHLF1(J)) + BO
ENDIF
ENDIF
RETURN
END

```

C
C
C
C
C
C
C
C

Subroutine Arrive

This subroutine takes a duein generated from the subroutine duein and determines when it will be delivered based CLIN delinquency report statistics for Medical. It will use the distribution developed to determine if an order will be early, on-time, 30 days late, 31-90 days late, 91 to 180 days late, or 181 + days late.

SUBROUTINE ARRIVE

```

DIMENSION QFD1(950,62),OHA1(950),ROP1(950),WRL1(950),P(950)
DIMENSION SL1(950),PC2(950),SHLF1(950),LT1(950)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1,P,SL1,PC,J,R,

```

```

*MONTHS, SDUEIN, PC2, SHLF1, DUEIN, IIT, LT1
R = RAND(IDUM)
IDUM = R*10000
IF (R .LE. .07) THEN
IIT = LT1(J) - 10
ELSE IF (R .GT. .07 .AND. R .LE. .66) THEN
IIT = LT1(J)
ELSE IF (R .GT. .66 .AND. R .LE. .78) THEN
IIT = 45 + LT1(J)
ELSE IF (R .GT. .78 .AND. R .LE. .88) THEN
IIT = 120 + LT1(J)
ELSE IF (R .GT. .88) THEN
IIT = 180 + LT1(J)
ELSE
ENDIF
RETURN
END

```

Subroutine Custdem

This subroutine forecasts customer demand at the beginning of each month/quarter using the double exponential smoothing technique utilized by the SAMMS system. The customer demand that is forecasted is then utilized in the stock replenishment process in the subroutine Stduein. The VIP items are looked at each month, with all others looked at each quarter.

```

SUBROUTINE CUSTDEM
CHARACTER*1 VIP1(950)
INTEGER TFACTOR, MONTHS, SHLF1(950)
DIMENSION QFD1(950,62), OHA1(950), ROP1(950), WRL1(950), SL1(950)
DIMENSION PC2(950), SYSS1(950,62), SYSD1(950,62), MAD1(950)
DIMENSION SHLF1(950), ALPHA1(950), SING(950,62), DOUB(950,62)
DIMENSION LT1(950), ARQCHK(950), PREDEM(950), FORDEM(950), P(950)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1, P, SL1, PC, J, R,
*MONTHS, SDUEIN, PC2, SHLF1, DUEIN, IIT, LT1
COMMON /BLK5/ISIM, SYSS1, SYSD1, MAD1, IT
COMMON /BLK6/ALPHA1, SING, DOUB
COMMON /BLK2/VIP1
COMMON /BLK3/ARQCHK, JK, TREC, PREDEM, FORDEM
IF ((VIP1(J).EQ. 'Y'.OR. VIP1(J).EQ. 'M').AND.ISIM.EQ. 1) THEN
SING(J,ISIM) = ALPHA1(J) * (PREDEM(J) - SYSS1(J,ISIM)) +
*SYSS1(J,ISIM)
DOUB(J,ISIM) = ALPHA1(J) * (SING(J,ISIM) - SYSD1(J,ISIM)) +
*SYSD1(J,ISIM)
QFD1(J,ISIM + 1) = 3.*(2.*SING(J,ISIM) - DOUB(J,ISIM))
IF (QFD1(J,ISIM + 1) .LT. 0.) THEN
QFD1(J,ISIM + 1) = QFD1(J,ISIM)
SING(J,ISIM) = QFD1(J,ISIM + 1)/3.
DOUB(J,ISIM) = QFD1(J,ISIM + 1)/3.
ELSE
ENDIF
GO TO 60
ELSE IF ((VIP1(J).EQ. 'Y'.OR. VIP1(J).EQ. 'M').AND. ISIM.EQ. 2)
*THEN

```



```

SING(J,ISIM) = ALPHA1(J) * (TREC - SING(J,ISIM - 1)) +
*SING(J,ISIM - 1)
DOUB(J,ISIM) = ALPHA1(J) * (SING(J,ISIM) - DOUB(J,ISIM - 1)) +
*DOUB(J,ISIM - 1)
QFD1(J,ISIM + 1) = 3.*(2.*SING(J,ISIM) - DOUB(J,ISIM))
IF (QFD1(J,ISIM + 1) .LT. 0.) THEN
QFD1(J,ISIM + 1) = (QFD1(J,ISIM - 1) + QFD1(J,1))/2.
SING(J,ISIM) = QFD1(J,ISIM + 1)/3.
DOUB(J,ISIM) = QFD1(J,ISIM + 1)/3.
ELSE
ENDIF
GO TO 60
ELSE IF ((VIP1(J) .EQ. 'Y'.OR. VIP1(J) .EQ. 'M').AND. ISIM
*.GT. 2) THEN
SING(J,ISIM) = ALPHA1(J) * (TREC - SING(J,ISIM - 1)) +
*SING(J,ISIM - 1)
DOUB(J,ISIM) = ALPHA1(J) * (SING(J,ISIM) - DOUB(J,ISIM - 1)) +
*DOUB(J,ISIM - 1)
QFD1(J,ISIM + 1) = 3.*(2.*SING(J,ISIM) - DOUB(J,ISIM))
IF (QFD1(J,ISIM + 1) .LT. 0.) THEN
QFD1(J,ISIM + 1) = (QFD1(J,ISIM - 2) + QFD1(J,ISIM - 1))/2.
SING(J,ISIM) = QFD1(J,ISIM + 1)/3.
DOUB(J,ISIM) = QFD1(J,ISIM + 1)/3.
ELSE
ENDIF
GO TO 60
ELSE IF (VIP1(J).EQ.'N'.AND. ISIM .EQ. 4) THEN
SING(J,ISIM) = ALPHA1(J) * (TREC - SYSS1(J,1)) + SYSS1(J,1)
DOUB(J,ISIM) = ALPHA1(J) * (SING(J,1) - SYSD1(J,1)) +
*SYSD1(J,ISIM)
FORDEM(J) = 2.*SING(J,ISIM) - DOUB(J,ISIM)
IF (FORDEM(J) .LT. 0.) THEN
FORDEM(J) = QFD1(J,1)
SING(J,ISIM) = FORDEM(J)
DOUB(J,ISIM) = FORDEM(J)
ELSE
ENDIF
JK = JK + 3
GO TO 60
ELSE IF (VIP1(J) .EQ. 'N' .AND. ISIM .EQ. JK) THEN
SING(J,ISIM) = ALPHA1(J) * (TREC - SING(J,ISIM - 3)) +
*SING(J,ISIM - 3)
DOUB(J,ISIM) = ALPHA1(J) * (SING(J,ISIM) - DOUB(J,ISIM - 3)) +
*DOUB(J,ISIM - 3)
FORDEM(J) = 2.*SING(J,ISIM) - DOUB(J,ISIM)
IF (FORDEM(J) .LT. 0.) THEN
IF (JK .EQ. 7) THEN
FORDEM(J) = (QFD1(J,1) + QFD1(J,ISIM - 3))/2.
ELSE
FORDEM(J) = (QFD1(J,ISIM - 6) + QFD1(J,ISIM - 3))/2.
ENDIF
SING(J,ISIM) = FORDEM(J)
DOUB(J,ISIM) = FORDEM(J)
ELSE

```

```

ENDIF
JK = JK + 3
GO TO 60
ENDIF
60 RETURN
END

```

C
C
C
C
C
C
C

Subroutine Requisition

This subroutine generates a requisition based on the items ARQ and the requisition quantity/ARQ probability curve developed by sampling requisitions and determining the ARQ of each NSN. The requisition generated is for a specific quantity as based in the items ARQ which was calculated using the previous 24 months of data.

```

SUBROUTINE REQN(BR, AY, R)
R = RAND(IDUM)
IDUM = R*10000
IF(R .LE. .93) THEN
AY = BR
ELSE IF(R .GT. .93 .AND. R .LE. .96) THEN
AY = BR * 2.
ELSE IF(R .GT. .96 .AND. R .LE. .975) THEN
AY = BR * 3.
ELSE IF(R .GT. .975 .AND. R .LE. .980) THEN
AY = BR * 4.
ELSE IF(R .GT. .98) THEN
AY = BR * 5.
ELSE
ENDIF
RETURN
END

```

C
C
C
C

Subroutine timer

Uses the routine itime for a seed for the random number generator

```

SUBROUTINE TIMER
COMMON IDUM, ISUM
IDUM = ITIME(IH,IM,IS)
ISUM = IH * IM * IS
IDUM = ISUM
RETURN
END

```

Appendix C

Computer Listing of the ARQ Simulation
Portion of the Model

```

C Program that runs a simulation model for the Medical
C Directorate. It will be used to test whether or not
C the Average Requisition Quantity (ARQ) concept of
C issuing stock should be continued. The ARQ concept states
C that if an IPG 2 requisition is 2X the ARQ, it goes on
C backorder, and if an IPG 3 requisition is 1.5X the ARQ, it
C goes on backorder. IPG 1 requisitions are always filled.
C
C ---THIS VERSION RUNS THE ARQ VERSION OF THE MODEL---
C ---AND IS BASED ON A 30 DAY MONTH
C
C ----THIS VERSION INCLUDES ORDERING STOCK TO COVER THE
C BACKORDERS GENERATED.
C
C The name of the program is arqbkor.f, and the output file is
C named works1
C
CHARACTER*1 ICC, VIP, NSN*13, NOM*30, NOM1(950)*30
CHARACTER*1 ICC1(950), VIP1(950), STAR, NSN1(950)*13
INTEGER YR, YS, TFACTOR, PCM, PCM1(950), ALT, PLT
INTEGER CTR, LT1(950), PC1(950), SHLF1(950), MONTHS, PC, SHLF
REAL INUM1(950), IBB1(60,30,50), IBB2(60,30,50)
REAL IBB3(60,30,50), IBB2A(60,30,50), IBB3A(60,30,50)
DIMENSION MRQ1(950), ARQ1(950), SPR1(950), IB(950), IZ(950)
DIMENSION IPG1(950), IPG2(950), IPG3(950), SUM1(950), FORDEM(950)
DIMENSION IBZ(950), IBZ1(950), IBZ2(950), IBZ3(950)
DIMENSION STOH(9100), REP(9100), IL(50), PREDEM(950)
DIMENSION IB1(950), IB2(950), IB3(950), PCP1(950), IREQS1(950)
DIMENSION IB2A(950), IB3A(950), TSTOCK(1850), REQOBJ(62)
DIMENSION IR1A(50), IR2A(50), IR3A(50), IB2A1(950), IB3A1(950)
DIMENSION BO1(60,30,50), BO2(60,30,50), BO3(60,30,50), SL1(950)
DIMENSION QFD1(950,62), OHA1(950), ROP1(950), WRL1(950), P(950)
DIMENSION SYSS1(950,62), SYSD1(950,62), MAD1(950), ALPHA1(950)
DIMENSION SING(950,62), DOUB(950,62), ARQCHK(950), PC2(950)
COMMON IDUM, ISUM
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1, P, SL1, PC, J, R,
*MONTHS, SDUEIN, PC2, SHLF1, DUEIN, IIT, LT1
COMMON /BLK5/ISIM, SYSS1, SYSD1, MAD1, IT
COMMON /BLK6/ALPHA1, SING, DOUB
COMMON /BLK2/VIP1
COMMON /BLK3/ARQCHK, JK, TREC, PREDEM, FORDEM
COMMON /BLK4/BO1, BO2, BO3, M1L, M2L, M3L, IR1A, IR2A, IR3A
IDUM = 100
R = RAND(IDUM)
WRITE(6, '("How many items will the model be using ?")')
READ(5,11) N
11 FORMAT(I5)
WRITE(6,12) N
12 FORMAT(/, 'The model is using', I5, ' items')
C
C
XSIM = 1100
NSX = 60
NF1 = NSX * 150

```

```

C The following is the set variable for the EOQ calculation
C
C     TFACTOR = 95
C
C The data files are now opened for use
C These files contain the data used in calculations in the
C simulation.
C
C     YR = 38
C     YS = 39
C     OPEN(YR,FILE='med1b')
C     OPEN(YS,FILE='med2b')
C     REWIND 38
C     REWIND 39
C     READ(YR,15)
C     READ(YS,15)
15 FORMAT(/,/ ,/,/,/,/,/,/,/,/)
C     JL = 74
C     OPEN(JL,FILE='response4')
C
C This is the loading of the data for the simulation and the
C beginning of the simulation itself.
C
C The variables are defined as follows:
C CTR--A counter used to identify each NSN
C NSN--The items NSN
C UP--The items unit price
C VIP--A key to determine if the item is a VIP item or not
C DV--The demand value code of an item
C ICC--The item's item category code
C MRQ--The item's maximum release quantity
C OHA--The on hand assets of the item at the beginning of the study
C ROP--The reorder point level of the item
C ARQ--The item's average requisition quantity
C QFD--The item's quarterly forecasted demand (It's monthly if VIP)
C SYSS--The item's system single exponential quantity
C SYSD--The item's system double exponential quantity
C SL--The item's safety level in days
C ALPHA--The item's alpha factor for exponential smoothing
C LT--The item's lead time in days
C PCM--The item's procurement lead time in months
C SHLF--The item's shelf life in months
C WRL--The item's war reserve protectable level
C SPR--The item's special program requirements-usually DEPMEDS
C MAD--The item's mean absolute deviation
C EEX--The item's historical average quarterly demand
C ESTD--The item's historical standard deviation of its EEX
C PCQ--The item's procurement cycle quantity--includes admin and plt
C PCPQ--The item's procurement cycle period expressed as a quantity.
C NOM--The item's nomenclature
C PC--The item's procurement cycle period expressed in months
C REQS--The item's number of requisitions per day
C INUM--The number of months an item has had demands

```

C SUM--The number of requisitions received during the months demanded
C

```
DO 20 IJ = 1, N
  READ(YR,100,END=50,ERR=900) CTR,NSN,NOM,VIP,QFD,SYSS,SYSD,
  *SL,ALPHA,ALT,PLT,PC,SHLF,MAD
100 FORMAT(1X,I5,1X,A13,1X,A19,2X,A1,10X,F7.0,1X,F8.1,1X,F8.1,
  *F7.0,2X,F5.2,2X,I3,2X,I3,2X,I3,4X,I2,2X,F8.1)
  NSN1(CTR) = NSN
  NOM1(CTR) = NOM
  VIP1(CTR) = VIP
  QFD1(CTR,1) = QFD
  SYSS1(CTR,1) = SYSS
  SYSD1(CTR,1) = SYSD
  SL1(CTR) = SL
  ALPHA1(CTR) = ALPHA
  LT1(CTR) = ALT + PLT
  PC1(CTR) = PC
  SHLF1(CTR) = SHLF
  MAD1(CTR) = MAD
  READ(YS,101,END=51,ERR=900) CTR,UP,MRQ,ICC,ROP,OHA,SPR,WRL,
  *SUM,ATOT,INUM,PCPQ,PCQ,PCM
101 FORMAT(1X,I5,14X,F8.2,1X,F6.0,4X,A1,F8.0,1X,F9.0,F8.0,F9.0,
  *4X,F6.0,3X,F8.2,4X,I3,2X,F8.0,1X,F8.0,I8)
  P(CTR) = UP
  MRQ1(CTR) = MRQ
  ICC1(CTR) = ICC
  ROP1(CTR) = ROP
  OHA1(CTR) = OHA
  SPR1(CTR) = SPR
  WRL1(CTR) = WRL
  SUM1(CTR) = SUM
  ARQ1(CTR) = ATOT
  INUM1(CTR) = INUM
  PCP1(CTR) = PCPQ
  PC2(CTR) = PCQ
  PCM1(CTR) = PCM
20 CONTINUE
50 CLOSE (YR)
51 CLOSE (YS)
  LF = N
```

C

```
WRITE(JL,733)
733 FORMAT(60X,'ARQ PROCESS RUN',//)
  WRITE(JL,700)
  WRITE(JL,701)
  WRITE(JL,702)
700 FORMAT(38X,'# REQS',4X,'AVE')
701 FORMAT(36X,'DURING LAST',1X,'REQ',3X,'MONTHS',6X,'NUMBER OF REQUIS
  *ITIONS',6X,'% TIMES',1X,'CUMULATIVE',6X,'BACKORDERS BY')
702 FORMAT(4X,'NSN',10X,'NOMENCLATURE',7X,'24 MONTHS',3X,'SIZE',1X,'DE
  *MANDED',3X,'IPG1',4X,'IPG2',4X,'IPG3',4X,'TOTAL',3X,
  *'NIS',3X,'BACKORDERS',4X,'IPG1',2X,'IPG2',2X,'IPG3',//)
```

```

C
C Trigger to print full output for an analysis of a particular
C NSN or group of NSNs. If KALT = 0, no print out is made, if
C KALT = 1, the full print out is made.
C
      KALT = 0
C
C Actual start of the simulation
C
      JJ = 65
      OPEN(JJ,FILE='CHECK4')
      DO 210 J = 1, N
      REQOBJ(J) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J) + PCP1(J)
      REQXXX = REQOBJ(J)
      IF(OHA1(J) .GT. REQOBJ(J)) THEN
      GO TO 55
      ELSE
      OHA1(J) = REQOBJ(J)
      ENDIF
55 TOTDUE = 0.
      IK = 0
      MA = 1
      IL(MA) = 0
      WARITE(JJ,650) NSN1(J), REQOBJ(J)
650 FORMAT('The model is processing NSN ',A13,' with requirements obj
      *=' ,F10.2)
      JK = 4
      UMMIPSA = SL1(J) * .8
      IF(VIP1(J) .EQ. 'Y' .OR. VIP1(J) .EQ. 'M') THEN
      UMMIPSB = QFD1(J,1)/2.
      ELSE
      UMMIPSB = QFD1(J,1)/6.
      ENDIF
      IF (UMMIPSA .LT. UMMIPSB) THEN
      UMMIPS = UMMIPSA
      ELSE
      UMMIPS = UMMIPSB
      ENDIF
      DO 120 LZ = 1, NF1
      STOH(LZ) = 0.
      REP(LZ) = 0.
120 CONTINUE
      DO 121 L1 = 1, 60
      DO 122 L2 = 1, 30
      DO 123 L3 = 1, 50
      IBB1(L1,L2,L3) = 0.
      IBB2(L1,L2,L3) = 0.
      IBB3(L1,L2,L3) = 0.
      IBB2A(L1,L2,L3) = 0.
      IBB3A(L1,L2,L3) = 0.
      BO1(L1,L2,L3) = 0.
      BO2(L1,L2,L3) = 0.
      BO3(L1,L2,L3) = 0.
123 CONTINUE

```

```

122 CONTINUE
121 CONTINUE
  IPG1(J) = 0
  IPG2(J) = 0
  IPG3(J) = 0
  IBZ(J) = 0
  IBZ1(J) = 0
  IBZ2(J) = 0
  IBZ3(J) = 0
  IB(J) = 0
  IB1(J) = 0
  IB2(J) = 0
  IB2A(J) = 0
  IB2A1(J) = 0
  IB3(J) = 0
  IB3A(J) = 0
  IB3A1(J) = 0
  IRPASS = 1
  LTOT = PC1(J)
  LTOT1 = PCM1(J)
  QFD1(J,1) = QFD1(J,1)
  SYSS1(J,1) = SYSS1(J,1)
  SYSD1(J,1) = SYSD1(J,1)
  IF (ROP1(J) .GT. OHA1(J)) THEN
    CALL STDUEIN
    CALL ARRIVE
    IK = IK + IIT
    TOTDUE = TOTDUE + DUEIN
    REP(IK) = DUEIN
  ELSE
    ENDIF
    IT = 1
    TSTOCK(IT) = OHA1(J)
    IT = 0
    RM = SUM1(J)/INUM1(J)
    RD = RM/30.
    IF(RD .LT. 1.) THEN
      LM = INT((1./RD) + .5)
      RZ = LM
      IRPASS = IRPASS + LM
      PREDEM(J) = (INT((30./RZ) + .5) * ARQ1(J))
      IREQS1(J) = 0
    ELSE
      IREQS1(J) = INT(RD + .5)
      PREDEM(J) = (IREQS1(J) * ARQ1(J)) * 30.
    ENDIF
    BL = 0.
    LB = 0
    TREC = 0.
    FORDEM(J) = 0.
    NUMREQ = IREQS1(J)

```

```

C
C Beginning of the ARQ process in the simulation
C

```



```

M1L = 1
M2L = 1
M3L = 1
DO 191 ISIM = 1, NSX
IF(VIP1(J) .EQ. 'N' .AND. ISIM .LE. 3) THEN
QFD1(J,ISIM) = QFD1(J,1)
ELSE
ENDIF
LB = LB + 1
IF((ISIM .LT. LTOT) .OR. (ISIM .LT. LTOT1)) THEN
REQOBJ(ISIM) = REQXXX
ELSE
ENDIF
IF(PC1(J) .EQ. PCM1(J)) GO TO 322

```

C
C This following loop takes care of the case when ltot = ltot1 = isim
C It uses the smaller of pc2 or pcpl to get more stock ordered
C

```

IF(LTOT .EQ. LTOT1 .AND. LTOT .EQ. ISIM) THEN
IF(PC2(J) .LT. PCP1(J)) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J)
LTOT = LTOT + PC1(J)
LTOT1 = LTOT1 + PCM1(J)
GO TO 323
ELSE
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PCP1(J)
LTOT = LTOT + PC1(J)
LTOT1 = LTOT1 + PCM1(J)
GO TO 323
ENDIF
ELSE
ENDIF

```

C
C End of the loop for the special case
C

```

IF(ISIM .EQ. LTOT) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J)
LTOT = LTOT + PC1(J)
GO TO 323
ELSE
ENDIF
IF (ISIM .EQ. LTOT1) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PCP1(J)
LTOT1 = LTOT1 + PCM1(J)
ELSE
ENDIF
GO TO 323
322 IF (ISIM .EQ. LTOT) THEN
REQOBJ(ISIM) = WRL1(J) + SPR1(J) + SL1(J) + PC2(J) + PCP1(J)
LTOT = LTOT + PC1(J)
ELSE
ENDIF
323 IF(VIP1(J) .EQ. 'N' .AND. LB .EQ. 4) THEN
FORDEM(J) = 0.

```

```

CALL CUSTDEM
TREC = 0.
LB = 1
ELSE
ENDIF
IF(VIP1(J) .EQ. 'N' .AND. ISIM .GT. 3) THEN
QFD1(J, ISIM) = FORDEM(J)
ELSE
ENDIF
IF(VIP1(J) .EQ. 'Y' .OR. VIP1(J) .EQ. 'M') THEN
CALL CUSTDEM
TREC = 0.
ELSE
ENDIF
ARQCHK(J) = QFD1(J, ISIM)/3.
DO 201 I = 1, 30
IT = IT + 1
TSTOCK(IT) = TSTOCK(IT) + REP(IT)
IF (KALT .EQ. 1) THEN
WRITE(JJ, 87) IT, TSTOCK(IT), REP(IT), BL, TOTDUE, JI, REP(JI), ISIM,
*REQOBJ (ISIM)
87 FORMAT(2X, I5, 4(3X, F10.2), 5X, I4, 5X, F10.2, 2X, I4, 2X, F10.2)
ELSE
ENDIF
IF(RD .LT. 1. .AND. IT .GT. 1) THEN
IF(IT .EQ. IRPASS) THEN
IRPASS = IRPASS + LM
GO TO 340
ELSE
ENDIF
ELSE
ENDIF
IF(RD .LT. 1. .AND. IT .EQ. 1) GO TO 131

```

C
C Beginning of the process to fill backorders in order of IPG group
C and time, i.e., when the backorder was generated
C

```

340 IF(TSTOCK(IT) .EQ. 0. .OR. IB(J) .EQ. 0 .OR. IT .LE. 300
*) GO TO 341
IF((IB1(J) - IBZ1(J)) .GT. 0) THEN
IF(RD .GT. 1.) THEN
IL1 = IREQS1(J)
ELSE
IL1 = M1L
ENDIF
DO 741 M1 = 11, ISIM
DO 850 K1 = 1, 30
DO 851 N1 = 1, IL1
IF(IBB1(M1, K1, IR1A(N1)) .EQ. 0. .OR. IR1A(N1) .EQ. 0)
*GO TO 979
IF(TSTOCK(IT) .LT. IBB1(M1, K1, IR1A(N1))) GO TO 341
TSTOCK(IT) = TSTOCK(IT) - IBB1(M1, K1, IR1A(N1))
IF(TSTOCK(IT) .EQ. 0.) THEN
IBZ(J) = IBZ(J) + 1

```

```

        IBZ1(J) = IBZ1(J) + 1
        TSTOCK(IT) = 0.
        BL = BL - IBB1(M1,K1,IR1A(N1))
        IBB1(M1,K1,IR1A(N1)) = 0.
        GO TO 341
    ELSE
        IBZ(J) = IBZ(J) + 1
        IBZ1(J) = IBZ1(J) + 1
        BL = BL - IBB1(M1,K1,IR1A(N1))
        IBB1(M1,K1,IR1A(N1)) = 0.
    ENDIF
979 CONTINUE
851 CONTINUE
850 CONTINUE
741 CONTINUE
    ELSE
    ENDIF
    IF(TSTOCK(IT) .LE. 0.) THEN
        TSTOCK(IT) = 0.
        GO TO 341
    ELSE
    ENDIF
    IF((IBZ(J) - IBZ2(J)) .GT. 0) THEN
        IF(RD .GT. 1.) THEN
            IL2 = IREQS1(J)
        ELSE
            IL2 = M2L
        ENDIF
        DO 742 M2 = 11, ISIM
        DO 852 K2 = 1, 30
        DO 853 N2 = 1, IL2
            IF(IBB2(M2,K2,IR2A(N2)) .EQ. 0.) GO TO 989
            IF(TSTOCK(IT) .LT. IBB2(M2,K2,IR2A(N2))) GO TO 341
            TSTOCK(IT) = TSTOCK(IT) - IBB2(M2,K2,IR2A(N2))
            IF(TSTOCK(IT) .EQ. 0.) THEN
                IBZ(J) = IBZ(J) + 1
                IBZ2(J) = IBZ2(J) + 1
                TSTOCK(IT) = 0.
                BL = BL - IBB2(M2,K2,IR2A(N2))
                IBB2(M2,K2,IR2A(N2)) = 0.
                GO TO 341
            ELSE
                IBZ(J) = IBZ(J) + 1
                IBZ2(J) = IBZ2(J) + 1
                BL = BL - IBB2(M2,K2,IR2A(N2))
                IBB2(M2,K2,IR2A(N2)) = 0.
            ENDIF
989 CONTINUE
853 CONTINUE
852 CONTINUE
742 CONTINUE
    ELSE
    ENDIF
    IF(TSTOCK(IT) .LE. 0.) THEN

```

```

TSTOCK(IT) = 0.
GO TO 341
ELSE
ENDIF
IF((IB3(J) - IBZ3(J)) .GT. 0) THEN
IF(RD .GT. 1.) THEN
IL3 = IREQS1(J)
ELSE
IL3 = M3L
ENDIF
DO 743 M3 = 11, ISIM
DO 854 K3 = 1, 30
DO 855 N3 = 1, IL3
IF(IBB3(M3,K3,IR3A(N3)) .EQ. 0.) GO TO 999
IF(TSTOCK(IT) .LT. IBB3(M3,K3,IR3A(N3))) GO TO 341
TSTOCK(IT) = TSTOCK(IT) - IBB3(M3,K3,IR3A(N3))
IF(TSTOCK(IT) .EQ. 0.) THEN
IBZ(J) = IBZ(J) + 1
IBZ3(J) = IBZ3(J) + 1
TSTOCK(IT) = 0.
BL = BL - IBB3(M3,K3,IR3A(N3))
IBB3(M3,K3,IR3A(N3)) = 0.
GO TO 341
ELSE
IBZ(J) = IBZ(J) + 1
IBZ3(J) = IBZ3(J) + 1
BL = BL - IBB3(M3,K3,IR3A(N3))
IBB3(M3,K3,IR3A(N3)) = 0.
ENDIF
999 CONTINUE
855 CONTINUE
854 CONTINUE
743 CONTINUE
ELSE
ENDIF

```

```

C
C End of the process to fill regularly generated backorders
C
C Beginning of process to fill backorders generated by the
C ARQ process. Will only fill when stock on hand is greater
C than the cut-off (30 days).
C

```

```

IF(TSTOCK(IT) .GT. ARQCHK(J) .AND. IT .GT. 300) THEN
IF(IB2A(J) .GT. 0) THEN
IF(RD .GT. 1.) THEN
IL2 = IREQS1(J)
ELSE
IL2 = M2L
ENDIF
DO 520 MB = 11, ISIM
DO 521 KB = 1, 30
DO 522 NB = 1, IL2
IF(IBB2A(MB,KB,IR2A(NB)) .EQ. 0.) GO TO 948
IF(TSTOCK(IT) .LT. ARQCHK(J)) GO TO 341

```

```

TSTOCK(IT) = TSTOCK(IT) - IBB2A(MB,KB,IR2A(NB))
IF(TSTOCK(IT) .GT. ARQCHK(J)) THEN
  IBZ(J) = IBZ(J) + 1
  IBZ2(J) = IBZ2(J) + 1
  BL = BL - IBB2A(MB,KB,IR2A(NB))
  IBB2A(MB,KB,IR2A(NB)) = 0.
  IB2A1(J) = IB2A1(J) + 1
ELSE IF(TSTOCK(IT) .EQ. ARQCHK(J)) THEN
  IBZ(J) = IBZ(J) + 1
  IBZ2(J) = IBZ2(J) + 1
  BL = BL - IBB2A(MB,KB,IR2A(NB))
  IBB2A(MB,KB,IR2A(NB)) = 0.
  IB2A1(J) = IB2A1(J) + 1
GO TO 341
ENDIF
948 CONTINUE
522 CONTINUE
521 CONTINUE
520 CONTINUE
ELSE
ENDIF
IF(TSTOCK(IT) .LE. ARQCHK(J)) GO TO 341
IF(IB3A(J) .GT. 0) THEN
  IF(RD .GT. 1.) THEN
    IL3 = IREQS1(J)
  ELSE
    IL3 = M3L
  ENDIF
  DO 523 MC = 11, ISIM
  DO 524 KC = 1, 30
  DO 525 NC = 1, IL3
  IF(IBB3A(MC,KC,IR3A(NC)) .EQ. 0.) GO TO 958
  IF(TSTOCK(IT) .LT. ARQCHK(J)) GO TO 341
  TSTOCK(IT) = TSTOCK(IT) - IBB3A(MC,KC,IR3A(NC))
  IF(TSTOCK(IT) .GT. ARQCHK(J)) THEN
    IBZ(J) = IBZ(J) + 1
    IBZ3(J) = IBZ3(J) + 1
    BL = BL - IBB3A(MC,KC,IR3A(NC))
    IBB3A(MC,KC,IR3A(NC)) = 0.
    IB3A1(J) = IB3A1(J) + 1
  ELSE IF(TSTOCK(IT) .EQ. ARQCHK(J)) THEN
    IBZ(J) = IBZ(J) + 1
    IBZ3(J) = IBZ3(J) + 1
    BL = BL - IBB3A(MC,KC,IR3A(NC))
    IBB3A(MC,KC,IR3A(NC)) = 0.
    IB3A1(J) = IB3A1(J) + 1
  GO TO 341
ENDIF
958 CONTINUE
525 CONTINUE
524 CONTINUE
523 CONTINUE
ELSE
ENDIF

```

```

        ELSE
        ENDIF
C
C End of the process to fill backorders generated by the ARQ
C process.
C
341 IF(RD .GT. 1.) THEN
    GO TO 149
    ELSE IF(IT .EQ. (IRPASS - LM)) THEN
    GO TO 131
    ELSE IF(RD .GT. .95 .AND. RD .LT. 1.05) THEN
    GO TO 131
    ELSE
    GO TO 127
    ENDIF
C
C Section that generates multiple requisitions per day, given
C that an NSN produces more than one requisition per day.
C
149 DO 199 L = 1, NUMREQ
    STAR = 1H
    IF(L .EQ. 1) THEN
    M1L = 1
    M2L = 1
    M3L = 1
    ELSE
    ENDIF
    CALL REQN(ARQ1(J), Y, R)
    TREC = TREC + Y
    IF (IT .GT. 300) THEN
    IA = R * 100
    IF (IA .LE. 9) IPG1(J) = IPG1(J) + 1
    IF (IA .GT. 9 .AND. IA .LE. 31) IPG2(J) = IPG2(J) + 1
    IF (IA .GT. 31) IPG3(J) = IPG3(J) + 1
    ELSE
    ENDIF
    IF (TSTOCK(IT) .EQ. 0. .AND. IT .GT. 300) THEN
    IB(J) = IB(J) + 1
    IF (IA .LE. 9) THEN
    IB1(J) = IB1(J) + 1
    IR1A(M1L) = L
    IBB1(ISIM,I,IR1A(M1L)) = Y
    BO1(ISIM,I,IR1A(M1L)) = Y
    BL = BL + IBB1(ISIM,I,IR1A(M1L))
    GO TO 129
    ELSE
    ENDIF
    IF (IA .GT. 9 .AND. IA .LE. 31) THEN
    IB2(J) = IB2(J) + 1
    IR2A(M2L) = L
    IBB2(ISIM,I,IR2A(M2L)) = Y
    BO2(ISIM,I,IR2A(M2L)) = Y
    BL = BL + IBB2(ISIM,I,IR2A(M2L))
    GO TO 129

```

```

ELSE
ENDIF
IF (IA .GT. 31) THEN
IB3(J) = IB3(J) + 1
IR3A(M3L) = L
IBB3(ISIM,I,IR3A(M3L)) = Y
BO3(ISIM,I,IR3A(M3L)) = Y
BL = BL + IBB3(ISIM,I,IR3A(M3L))
GO TO 129
ELSE
ENDIF
ELSE IF(TSTOCK(IT) .LE. ARQCHK(J) .AND. IT .GT. 300) THEN
IF(IA .LE. 9) THEN
GO TO 703
ELSE IF((IA .GT. 9 .AND. IA .LE. 31) .AND. Y .GE. (2.*ARQ1(J))
*) THEN
STAR = 1H*
IB(J) = IB(J) + 1
IB2(J) = IB2(J) + 1
IR2A(M2L) = L
IBB2A(ISIM,I,IR2A(M2L)) = Y
BO2(ISIM,I,IR2A(M2L)) = Y
BL = BL + IBB2A(ISIM,I,IR2A(M2L))
IB2A(J) = IB2A(J) + 1
GO TO 129
ELSE IF(IA .GT. 31 .AND. Y .GE. (1.5*ARQ1(J))) THEN
STAR = 1H*
IB(J) = IB(J) + 1
IB3(J) = IB3(J) + 1
IR3A(M3L) = L
IBB3A(ISIM,I,IR3A(M3L)) = Y
BO3(ISIM,I,IR3A(M3L)) = Y
BL = BL + IBB3A(ISIM,I,IR3A(M3L))
IB3A(J) = IB3A(J) + 1
GO TO 129
ENDIF
GO TO 791
ENDIF
GO TO 793
703 IF(TSTOCK(IT) .GE. Y) THEN
TSTOCK(IT) = TSTOCK(IT) - Y
IF(TSTOCK(IT) .LE. 0.) TSTOCK(IT) = 0.
GO TO 129
ELSE
IB(J) = IB(J) + 1
IB1(J) = IB1(J) + 1
IR1A(M1L) = L
IBB1(ISIM,I,IR1A(M1L)) = Y
BO1(ISIM,I,IR1A(M1L)) = Y
BL = BL + IBB1(ISIM,I,IR1A(M1L))
GO TO 129
ENDIF
791 IF((IA .GT. 9 .AND. IA .LE. 31) .AND. TSTOCK(IT) .LT. Y
*) THEN

```

```

IB(J) = IB(J) + 1
IB2(J) = IB2(J) + 1
IR2A(M2L) = L
IBB2(ISIM,I,IR2A(M2L)) = Y
BO2(ISIM,I,IR2A(M2L)) = Y
BL = BL + IBB2(ISIM,I,IR2A(M2L))
GO TO 129
ELSE IF(IA .GT. 31 .AND. TSTOCK(IT) .LT. Y) THEN
IB(J) = IB(J) + 1
IB3(J) = IB3(J) + 1
IR3A(M3L) = L
IBB3(ISIM,I,IR3A(M3L)) = Y
BO3(ISIM,I,IR3A(M3L)) = Y
BL = BL + IBB3(ISIM,I,IR3A(M3L))
GO TO 129
ENDIF
793 TSTOCK(IT) = TSTOCK(IT) - Y
IF(TSTOCK(IT) .LE. 0.) TSTOCK(IT) = 0.
129 M1L = M1L + 1
M2L = M2L + 1
M3L = M3L + 1
STAR = 1H
199 CONTINUE
TSTOCK(IT + 1) = TSTOCK(IT)
IF(TSTOCK(IT + 1) .LE. 0.) TSTOCK(IT + 1) = 0.

```

C
C End of the multiple requisition process
C

GO TO 128

C
C Process to generate a single requisition per day, if only one per
C NSN is required
C

```

131 CALL REQN(ARQ1(J), Y, R)
TREC = TREC + Y
IF (IT .GT. 300) THEN
IA = R * 100
IF (IA .LE. 9) IPG1(J) = IPG1(J) + 1
IF (IA .GT. 9 .AND. IA .LE. 31) IPG2(J) = IPG2(J) + 1
IF (IA .GT. 31) IPG3(J) = IPG3(J) + 1
ELSE
ENDIF
IF (TSTOCK(IT) .EQ. 0. .AND. IT .GT. 300) THEN
IB(J) = IB(J) + 1
IF (IA .LE. 9) THEN
IB1(J) = IB1(J) + 1
IR1A(M1L) = 1
IBB1(ISIM,I,IR1A(M1L)) = Y
BO1(ISIM,I,IR1A(M1L)) = Y
BL = BL + IBB1(ISIM,I,IR1A(M1L))
GO TO 127
ELSE
ENDIF
IF (IA .GT. 9 .AND. IA .LE. 31) THEN

```



```

IB2(J) = IB2(J) + 1
IR2A(M2L) = 1
IBB2(ISIM,I,IR2A(M2L)) = Y
BO2(ISIM,I,IR2A(M2L)) = Y
BL = BL + IBB2(ISIM,I,IR2A(M2L))
GO TO 127
ELSE
ENDIF
IF (IA .GT. 31) THEN
IB3(J) = IB3(J) + 1
IR3A(M3L) = 1
IBB3(ISIM,I,IR3A(M3L)) = Y
BO3(ISIM,I,IR3A(M3L)) = Y
BL = BL + IBB3(ISIM,I,IR3A(M3L))
GO TO 127
ELSE
ENDIF
ELSE IF(TSTOCK(IT) .LE. ARQCHK(J) .AND. IT .GT. 300) THEN
IF(IA .LE. 9) THEN
GO TO 704
ELSE IF((IA .GT. 9 .AND. IA .LE. 31).AND. Y.GE. (2.*ARQ1(J))
*) THEN
IB(J) = IB(J) + 1
IB2(J) = IB2(J) + 1
IR2A(M2L) = 1
IBB2A(ISIM,I,IR2A(M2L)) = Y
BO2(ISIM,I,IR2A(M2L)) = Y
BL = BL + IBB2A(ISIM,I,IR2A(M2L))
IB2A(J) = IB2A(J) + 1
GO TO 127
ELSE IF(IA .GT. 31 .AND. Y .GE. (1.5*ARQ1(J))) THEN
IB(J) = IB(J) + 1
IB3(J) = IB3(J) + 1
IR3A(M3L) = 1
IBB3A(ISIM,I,IR3A(M3L)) = Y
BO3(ISIM,I,IR3A(M3L)) = Y
BL = BL + IBB3A(ISIM,I,IR3A(M3L))
IB3A(J) = IB3A(J) + 1
GO TO 127
ENDIF
GO TO 794
ENDIF
GO TO 795
704 IF(TSTOCK(IT) .GE. Y) THEN
TSTOCK(IT + 1) = TSTOCK(IT) - Y
IF(TSTOCK(IT + 1) .LE. 0.) TSTOCK(IT + 1) = 0.
GO TO 128
ELSE
IB(J) = IB(J) + 1
IB1(J) = IB1(J) + 1
IR1A(M1L) = 1
IBB1(ISIM,I,IR1A(M1L)) = Y
BO1(ISIM,I,IR1A(M1L)) = Y
BL = BL + IBB1(ISIM,I,IR1A(M1L))

```

```

        GO TO 127
    ENDIF
794 IF((IA.GT. 9 .AND. IA .LE. 31).AND. TSTOCK(IT) .LT. Y
*) THEN
    IB(J) = IB(J) + 1
    IB2(J) = IB2(J) + 1
    IR2A(M2L) = 1
    IBB2(ISIM,I,IR2A(M2L)) = Y
    BO2(ISIM,I,IR2A(M2L)) = Y
    BL = BL + IBB2(ISIM,I,IR2A(M2L))
    GO TO 127
ELSE IF(IA .GT. 31 .AND. TSTOCK(IT) .LT. Y) THEN
    IB(J) = IB(J) + 1
    IB3(J) = IB3(J) + 1
    IR3A(M3L) = 1
    IBB3(ISIM,I,IR3A(M3L)) = Y
    BO3(ISIM,I,IR3A(M3L)) = Y
    BL = BL + IBB3(ISIM,I,IR3A(M3L))
    GO TO 127
ENDIF
795 TSTOCK(IT + 1) = TSTOCK(IT) - Y
    IF(TSTOCK(IT + 1) .LE. 0.) TSTOCK(IT + 1) = 0.
    GO TO 128
127 TSTOCK(IT + 1) = TSTOCK(IT)
C
C End of the requisition generating process
C
128 IF (IT .EQ. IK) THEN
    TOTDUE = TOTDUE - REP(IT)
ELSE
    ENDIF
DO 251 KJ = 1,MA
    IF(IT .EQ. IL(KJ)) THEN
        TOTDUE = TOTDUE - REP(IL(KJ))
        GO TO 171
    ELSE
        ENDIF
251 CONTINUE
171 CONTINUE
C
C Section to check and see if an order needs to be made
C
    IF((TSTOCK(IT + 1) + TOTDUE - BL) .LE. REQOBJ(ISIM)) THEN
        CALL STDUE(RD,IREQS1(J))
        CALL ARRIVE
        JI = IT + IIT
        IF(IL(MA) .EQ. JI) THEN
            REP(JI) = REP(JI) + SDUEIN
            GO TO 181
        ELSE
            ENDIF
        MA = MA + 1
        IL(MA) = JI
        REP(JI) = SDUEIN

```

```

181 TOTDUE = TOTDUE + SDUEIN
    ELSE
    ENDIF
201 CONTINUE
191 CONTINUE
210 CONTINUE
    JA = 0
    JA1 = 0
    JA2 = 0
    JA2A = 0
    JA2A1 = 0
    JA3 = 0
    JA3A = 0
    JA3A1 = 0
    ISUM1 = 0
    ISUM2 = 0
    ISUM3 = 0
    IGTOT = 0
    LA = 0
    LA1 = 0
    LA2 = 0
    LA3 = 0
    DO 303 K = 1, LF
        JA = JA + IB(K)
        JA1 = JA1 + IB1(K)
        JA2 = JA2 + IB2(K)
        JA2A = JA2A + IB2A(K)
        JA2A1 = JA2A1 + IB2A1(K)
        JA3 = JA3 + IB3(K)
        JA3A = JA3A + IB3A(K)
        JA3A1 = JA3A1 + IB3A1(K)
        ISUM1 = ISUM1 + IPG1(K)
        ISUM2 = ISUM2 + IPG2(K)
        ISUM3 = ISUM3 + IPG3(K)
        LA = LA + IBZ(K)
        LA1 = LA1 + IBZ1(K)
        LA2 = LA2 + IBZ2(K)
        LA3 = LA3 + IBZ3(K)
        IZ(K) = IPG1(K) + IPG2(K) + IPG3(K)
        AB = IZ(K)
        BB = (IB(K)/AB)*100.
        WRITE(JL,504)NSN1(K),NOM1(K),SUM1(K),ARQ1(K),INUM1(K),IPG1(K),IP
        *G2(K),IPG3(K),IZ(K),BB,IB(K),IB1(K),IB2(K),IB3(K)
504 FORMAT(A13,2X,A22,F6.0,1X,F8.2,3X,F3.0,5X,I4,4X,I4,4X,I5,3X,I6,
        *2X,F6.2,1X,I6,6X,I5,1X,I5,2X,I5)
303 CONTINUE
        IGTOT = IGTOT + ISUM1 + ISUM2 + ISUM3
        XSUM1 = ISUM1
        XSUM2 = ISUM2
        XSUM3 = ISUM3
        XIGTOT = IGTOT
        P1 = (XSUM1/XIGTOT)*100.
        P2 = (XSUM2/XIGTOT)*100.
        P3 = (XSUM3/XIGTOT)*100.

```

```

WA = (JA/XIGTOT)*100.
WA1 = ((JA - LA)/XIGTOT)*100.
IWA2 = JA - LA
IWA3 = JA1 - LA1
IWA4 = JA2 - LA2
IWA5 = JA3 - LA3
WRITE(JL,508) ISUM1, ISUM2, ISUM3, IGTOT, JA, JA1, JA2, JA3, P1, P2, P3
508 FORMAT(/,54X, 'TOTALS', 1X, I6, 2X, I6, 3X, I6, 3X, I6, 9X, I6, 6X, I5, 1X, I
*5, 2X, I5, /, 50X, '% OF TOTAL', 2X, F5.2, 3X, F5.2, 4X, F5.2)
WRITE(JL,509) WA
509 FORMAT(/,74X, 'WEIGHTED AVG NIS', 6X, F5.2)
WRITE(JL,528) LA, LA1, LA2, LA3
528 FORMAT(/,73X, 'BACKORDERS FILLED', 12X, I6, 6X, I5, 1X, I5, 2X, I5)
WRITE(JL,529) WA1, IWA2, IWA3, IWA4, IWA5
529 FORMAT(/,52X, 'REVISED NIS (AFTER FILLING BACKORDERS)', 6X, F5.2, 1X,
*I6, 6X, I5, 1X, I5, 2X, I5)
WRITE(JL,510) JA2A, JA3A
510 FORMAT(/,73X, 'BACKORDERS RESULTING FROM THE ARQ PROCESS', 6X, I5, 2
*X, I5)
WRITE(JL,512) JA2A1, JA3A1
512 FORMAT(/,73X, 'ARQ IPG2 AND IPG3 BACKORDERS FILLED', 12X, I5, 2X, I5)
WRITE(JL,511)
511 FORMAT(/,73X, 'PROCESS MODIFIED WITH IPG2 REQS TRIGGERED AT 2X ARQ
*')
STOP
900 WRITE(6, '("ERROR ENCOUNTERED")')
END

```

C
C
C
C
C
C
C

Subroutine Stduein

Is the seed for the model. It tests if the current on-hand assets are at a reorder point (if so it generates an order), or if they are sufficient to begin the simulation. If an order is generated, the model uses the SAMMS criteria outlined in the EOQ process.

```

SUBROUTINE STDUEIN
INTEGER SHLF1(950), MONTHS, TFACTOR
DIMENSION QFD1(950,62), OHA1(950), ROP1(950), P(950), PC2(950)
DIMENSION WRL1(950), SL1(950), SHLF1(950), LT1(950)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1, P, SL1, PC, J, R,
*MONTHS, SDUEIN, PC2, SHLF1, DUEIN, IIT, LT1
DVQD = QFD1(J,1) * P(J)
IF (DVQD .LE. 62. .AND. SHLF1(J) .GE. 36) THEN
DUEIN = 12. * QFD1(J,1)
ELSE IF (DVQD .LE. 62. .AND. SHLF1(J) .LT. 36) THEN
IF (SHLF1(J) .EQ. 0) THEN
DUEIN = PC2(J)
ELSE
DUEIN = (QFD1(J,1)/3.) * SHLF1(J)
ENDIF
ELSE IF (DVQD .GT. 62. .AND. DVQD .LE. 1846.) THEN
EOQ = TFACTOR * (SQRT(DVQD))
MONTHS = (EOQ/P(J))/(QFD1(J,1)/3.)
IF (SHLF1(J) .LT. MONTHS) THEN
IF (SHLF1(J) .EQ. 0) THEN

```

```

DUEIN = PC2(J)
ELSE
XX = SHLF1(J)
XXN = MONTHS
DUEIN = (EOQ/P(J)) * (XX/XXN)
ENDIF
ELSE
DUEIN = EOQ/P(J)
ENDIF
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.GE. 9) THEN
DUEIN = 3. * QFD1(J,1)
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.LT. 9) THEN
IF (SHLF1(J) .EQ. 0) THEN
DUEIN = PC2(J)
ELSE
DUEIN = (QFD1(J,1)/3.) * SHLF1(J)
ENDIF
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .GE. 6) THEN
DUEIN = 2. * QFD1(J,1)
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .LT. 6) THEN
IF (SHLF1(J) .EQ. 0) THEN
DUEIN = PC2(J)
ELSE
DUEIN = (QFD1(J,1)/3.) * SHLF1(J)
ENDIF
ENDIF
RETURN
END

```

C
C
C
C
C
C
C

Subroutine Stdue

Is used in the simulation to test if stock has fallen below its reorder point (if so it generates an order). If an order is generated, the simulation uses the SAMMS criteria outlined in the EOQ process.

```

SUBROUTINE STDUE(XRD,KIR)
INTEGER SHLF1(950), MONTHS, TFACTOR
DIMENSION QFD1(950,62), OHA1(950), ROP1(950),P(950),PC2(950)
DIMENSION WRL1(950),SL1(950),SHLF1(950),LT1(950)
DIMENSION SYSS1(950,62),SYSD1(950,62),MAD1(950)
DIMENSION BO1(60,30,50), BO2(60,30,50), BO3(60,30,50)
DIMENSION IR1A(50), IR2A(50), IR3A(50)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1,P,SL1,PC,J,R,
*MONTHS, SDUEIN, PC2,SHLF1,DUEIN,IIT,LT1
COMMON /BLK5/ISIM, SYSS1, SYSD1, MAD1,IT
COMMON /BLK4/BO1, BO2, BO3, M1L, M2L,M3L,IR1A,IR2A,IR3A
BO = 0.
IF(IT .GT. 300) THEN
IF(XRD .GT. 1.) THEN
KL1 = KIR
KL2 = KIR
KL3 = KIR

```

```

ELSE
KL1 = M1L
KL2 = M2L
KL3 = M3L
ENDIF
DO 25 KZ = 11, ISIM
DO 30 KA = 1, 30
DO 35 LP = 1, KL1
IF(IR1A(LP) .EQ. 0) GO TO 35
BO = BO + BO1(KZ,KA,IR1A(LP))
BO1(KZ,KA,IR1A(LP)) = 0.
35 CONTINUE
30 CONTINUE
25 CONTINUE
DO 26 JE = 11, ISIM
DO 31 KE = 1, 30
DO 36 LE = 1, KL2
BO = BO + BO2(JE,KE,IR2A(LE))
BO2(JE,KE,IR2A(LE)) = 0.
36 CONTINUE
31 CONTINUE
26 CONTINUE
DO 27 JD = 11, ISIM
DO 32 KD = 1, 30
DO 37 LD = 1, KL3
BO = BO + BO3(JD,KD,IR3A(LD))
BO3(JD,KD,IR3A(LD)) = 0.
37 CONTINUE
32 CONTINUE
27 CONTINUE
ELSE
ENDIF
DVQD = QFD1(J,ISIM) * P(J)
IF (DVQD .LE. 62. .AND. SHLF1(J) .GE. 36) THEN
SDUEIN = 12. * QFD1(J,ISIM) + BO
ELSE IF (DVQD .LE. 62. .AND. SHLF1(J) .LT. 36) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
SDUEIN = (QFD1(J,ISIM)/3.) * SHLF1(J) + BO
ENDIF
ELSE IF (DVQD .GT. 62. .AND. DVQD .LE. 1846.) THEN
EOQ = TFACTOR * (SQRT(DVQD))
MONTHS = (EOQ/P(J))/(QFD1(J,ISIM)/3.)
IF (SHLF1(J) .LT. MONTHS) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
XX = SHLF1(J)
XXN = MONTHS
SDUEIN = (EOQ/P(J)) * (XX/XXN) + BO
ENDIF
ELSE
SDUEIN = EOQ/P(J) + BO

```

```

ENDIF
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.GE. 9) THEN
SDUEIN = 3. * QFD1(J,ISIM) + BO
ELSE IF (DVQD .GT. 1846. .AND. DVQD .LE. 15000. .AND. SHLF1(J)
*.LT. 9) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
SDUEIN = (QFD1(J,ISIM)/3.) * SHLF1(J) + BO
ENDIF
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .GE. 6) THEN
SDUEIN = 2. * QFD1(J,ISIM) + BO
ELSE IF (DVQD .GT. 15000. .AND. SHLF1(J) .LT. 6) THEN
IF (SHLF1(J) .EQ. 0) THEN
SDUEIN = PC2(J) + BO
ELSE
SDUEIN = (QFD1(J,ISIM)/3.) * SHLF1(J) + BO
ENDIF
ENDIF
RETURN
END

```

C

C Subroutine Arrive

C This subroutine takes a duein generated from the subroutine duein
C and determines when it will be delivered based CLIN delinquency
C report statistics for Medical. It will use the distribution
C developed to determine if an order will be early, on-time, 30 days
C late, 31-90 days late, 91 to 180 days late, or 181 + days late.

C

```

SUBROUTINE ARRIVE
DIMENSION QFD1(950,62), OHA1(950), ROP1(950),P(950),PC2(950)
DIMENSION WRL1(950),SL1(950),SHLF1(950),LT1(950)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1,P,SL1,PC,J,R,
*MONTHS, SDUEIN, PC2,SHLF1,DUEIN,IIT,LT1
R = RAND(IDUM)
IDUM = R*10000
IF (R .LE. .07) THEN
IIT = LT1(J) - 10
ELSE IF (R .GT. .07 .AND. R .LE. .66) THEN
IIT = LT1(J)
ELSE IF (R .GT. .66 .AND. R .LE. .78) THEN
IIT = 45 + LT1(J)
ELSE IF (R .GT. .78 .AND. R .LE. .88) THEN
IIT = 120 + LT1(J)
ELSE IF (R .GT. .88) THEN
IIT = 180 + LT1(J)
ELSE
ENDIF
RETURN
END

```

C

C Subroutine Custdem

C This subroutine forecasts customer demand at the beginning of each

C month/quarter using the double exponential smoothing technique
C utilized by the SAMMS system. The customer demand that is forecasted
C is then utilized in the stock replenishment process in the
C subroutine Stduein. The VIP items are looked at each month, with all
C others looked at each quarter.
C

```

SUBROUTINE CUSTDEM
CHARACTER*1 VIP1(950)
INTEGER TFACTOR, MONTHS, SHLF1(950)
DIMENSION QFD1(950,62), OHA1(950), ROP1(950), P(950), PC2(950)
DIMENSION WRL1(950), SL1(950), SHLF1(950), LT1(950), ALPHA1(950)
DIMENSION SYSS1(950,62), SYSD1(950,62), MAD1(950), FORDEM(950)
DIMENSION SING(950,62), DOUB(950,62), ARQCHK(950), PREDEM(950)
COMMON /BLK1/QFD1, TFACTOR, OHA1, ROP1, WRL1, P, SL1, PC, J, R,
*MONTHS, SDUEIN, PC2, SHLF1, DUEIN, IIT, LT1
COMMON /BLK5/ISIM, SYSS1, SYSD1, MAD1, IT
COMMON /BLK6/ALPHA1, SING, DOUB
COMMON /BLK2/VIP1
COMMON /BLK3/ARQCHK, JK, TREC, PREDEM, FORDEM
IF ((VIP1(J).EQ. 'Y'.OR. VIP1(J).EQ. 'M').AND. ISIM.EQ. 1) THEN
SING(J, ISIM) = ALPHA1(J) * (PREDEM(J) - SYSS1(J, ISIM)) +
*SYSS1(J, ISIM)
DOUB(J, ISIM) = ALPHA1(J) * (SING(J, ISIM) - SYSD1(J, ISIM)) +
*SYSD1(J, ISIM)
QFD1(J, ISIM + 1) = 3.*(2.*SING(J, ISIM) - DOUB(J, ISIM))
IF (QFD1(J, ISIM + 1) .LT. 0.) THEN
QFD1(J, ISIM + 1) = QFD1(J, ISIM)
SING(J, ISIM) = QFD1(J, ISIM + 1)/3.
DOUB(J, ISIM) = QFD1(J, ISIM + 1)/3.
ELSE
ENDIF
GO TO 60
ELSE IF ((VIP1(J).EQ. 'Y'.OR. VIP1(J).EQ. 'M').AND. ISIM.EQ. 2)
*THEN
SING(J, ISIM) = ALPHA1(J) * (TREC - SING(J, ISIM - 1)) +
*SING(J, ISIM - 1)
DOUB(J, ISIM) = ALPHA1(J) * (SING(J, ISIM) - DOUB(J, ISIM - 1)) +
*DOUB(J, ISIM - 1)
QFD1(J, ISIM + 1) = 3.*(2.*SING(J, ISIM) - DOUB(J, ISIM))
IF (QFD1(J, ISIM + 1) .LT. 0.) THEN
QFD1(J, ISIM + 1) = (QFD1(J, ISIM - 1) + QFD1(J, 1))/2.
SING(J, ISIM) = QFD1(J, ISIM + 1)/3.
DOUB(J, ISIM) = QFD1(J, ISIM + 1)/3.
ELSE
ENDIF
GO TO 60
ELSE IF ((VIP1(J).EQ. 'Y'.OR. VIP1(J).EQ. 'M').AND. ISIM
*.GT. 2) THEN
SING(J, ISIM) = ALPHA1(J) * (TREC - SING(J, ISIM - 1)) +
*SING(J, ISIM - 1)
DOUB(J, ISIM) = ALPHA1(J) * (SING(J, ISIM) - DOUB(J, ISIM - 1)) +
*DOUB(J, ISIM - 1)
QFD1(J, ISIM + 1) = 3.*(2.*SING(J, ISIM) - DOUB(J, ISIM))
IF (QFD1(J, ISIM + 1) .LT. 0.) THEN

```



```

QFD1(J,ISIM + 1) = (QFD1(J,ISIM - 2) + QFD1(J,ISIM - 1))/2.
SING(J,ISIM) = QFD1(J,ISIM + 1)/3.
DOUB(J,ISIM) = QFD1(J,ISIM + 1)/3.
ELSE
ENDIF
GO TO 60
ELSE IF (VIP1(J) .EQ. 'N' .AND. ISIM .EQ. 4) THEN
SING(J,ISIM) = ALPHA1(J) * (TREC - SYSS1(J,1)) + SYSS1(J,1)
DOUB(J,ISIM) = ALPHA1(J) * (SING(J,1) - SYSD1(J,1)) +
*SYSD1(J,ISIM)
FORDEM(J) = 2.*SING(J,ISIM) - DOUB(J,ISIM)
IF (FORDEM(J) .LT. 0.) THEN
FORDEM(J) = QFD1(J,1)
SING(J,ISIM) = FORDEM(J)
DOUB(J,ISIM) = FORDEM(J)
ELSE
ENDIF
JK = JK + 3
GO TO 60
ELSE IF (VIP1(J) .EQ. 'N' .AND. ISIM .EQ. JK) THEN
SING(J,ISIM) = ALPHA1(J) * (TREC - SING(J,ISIM - 3)) +
*SING(J,ISIM - 3)
DOUB(J,ISIM) = ALPHA1(J) * (SING(J,ISIM) - DOUB(J,ISIM - 3)) +
*DOUB(J,ISIM - 3)
FORDEM(J) = 2.*SING(J,ISIM) - DOUB(J,ISIM)
IF (FORDEM(J) .LT. 0.) THEN
IF (JK .EQ. 7) THEN
FORDEM(J) = (QFD1(J,1) + QFD1(J,ISIM - 3))/2.
ELSE
FORDEM(J) = (QFD1(J,ISIM - 6) + QFD1(J,ISIM - 3))/2.
ENDIF
SING(J,ISIM) = FORDEM(J)
DOUB(J,ISIM) = FORDEM(J)
ELSE
ENDIF
JK = JK + 3
GO TO 60
ENDIF
60 RETURN
END

```

```

C
C Subroutine Requisition
C This subroutine generates a requisition based on the items ARQ
C and the requisition quantity/ARQ probability curve developed by
C sampling requisitions and determining the ARQ of each NSN. The
C requisition generated is for a specific quantity as based in the
C items ARQ which was calculated using the previous 24 months of data.
C

```

```

SUBROUTINE REQN(BR, AY, R)
R = RAND(IDUM)
IDUM = R*10000
IF(R .LE. .93) THEN
AY = BR
ELSE IF(R .GT. .93 .AND. R .LE. .96) THEN

```

```

    AY = BR * 2.
    ELSE IF(R .GT. .96 .AND. R .LE. .975) THEN
    AY = BR * 3.
    ELSE IF(R .GT. .975 .AND. R .LE. .980) THEN
    AY = BR * 4.
    ELSE IF(R .GT. .98) THEN
    AY = BR * 5.
    ELSE
    ENDIF
    RETURN
    END

```

```

C
C Subroutine timer
C Uses the routine itime for a seed for the random number generator
C

```

```

    SUBROUTINE TIMER
    COMMON IDUM, ISUM
    IDUM = ITIME(IH,IM,IS)
    ISUM = IH * IM * IS
    IDUM = ISUM
    RETURN
    END

```

Appendix D

Sensitivity Analysis Results

SENSITIVITY ANALYSIS RESULTS

Sensitivity analyses were conducted on both the SAMMS and Requisition Optimization segments of the model to compare the results of a number of different 'what if' scenarios with the results originally obtained. The original probability distribution utilized to generate demands was reconstructed and used in these analyses. The results of the four areas analyzed are:

a. Reconstruction of the probability distribution function of requisitions/average requisition size.

Individual requisition quantities are simulated by the model based upon an analysis of 4th quarter FY 88 requisitions contained in the DLA Integrated Data Bank. To obtain these requisitions, a random selection of 10% of the status code 1 and 7 items was made. This produced a sample of 1,279 NSNs. The reconstruction of the probability distribution included analyzing the 41,796 requisitions produced by these 1,279 NSN's. For each NSN randomly selected, its average requisition quantity (ARQ) was calculated. Then, each NSN's ARQ was matched against its requisition(s) from the sample of 41,796 requisitions. The requisition size was divided by the item's ARQ and totals were kept for multiples of one to sixteen. For each multiple, the demand of each requisition was also totaled. The resulting distribution is provided as figure 1. The differences from the original distribution result from having the information on the 41,796 requisitions, versus estimating what it was.

b. Rerunning the model with the reconstructed data.

Both the SAMMS portion and the Requisition Optimization portion of the model were rerun with the revised probability distribution function of requisitions/average requisition size. When demands were generated by the model, multiples of up to sixteen (16) times the ARQ were included. This insured that the demands generated by the model were consistent with the percentages outlined on the probability distribution shown on figure 1. The results obtained by the SAMMS portion and the Requisition Optimization portion were identical to those originally reported. Over the five years of simulation, the requisition optimization policy proved to be no better or worse than the current SAMMS policy. Figure 2 provides the results.

c. Provide a sensitivity analysis of how the 874 NSN's simulated are influenced by:

- (1) Erratic requisition patterns.

In the original model, if an item received multiple requisitions per day, that number was held constant. Varying this number would provide simulation results more in line with what actually happens. Results of these runs indicate that the Requisition Optimization process is more sensitive to the variability in requisitions received per day and therefore should probably be used on items experiencing such high requisition variability. Figure 3 provides the results.

(2) Runs stratified by Federal Stock Category (FSC).

The four major federal stock categories in Medical, Drugs (6505), Surgical Equipment (6510,6515), X-ray Dental and Hospital Equipment (6525,6520,6530), and Optical Equipment and Lenses (6540) were analyzed using the new probability distribution function. As noted in Figures 4-7, the following results were obtained:

(A) Drugs: The requisition optimization policy is approximately 2% better than SAMMS, and produces fewer IPG 1,2, and 3 backorders.

(B) Surgical Equipment: There was little difference between policies in terms of supply availability, with the requisition optimization policy producing fewer IPG 2 backorders, but more IPG 3 backorders.

(C) X-Ray/Dental/Hospital Equipment: Again, there was little difference between the policies in terms of supply availability, with the requisition optimization policy producing fewer IPG 2 backorders, but more IPG 3 backorders.

(D) Optical Equipment and Lenses: This was the only category where the SAMMS policy produced better results than the requisition optimization policy. It is felt that this is due to the very erratic requisition and demand patterns of these items.

The analysis of the FSC categories indicated that, with the exception of Drugs and Optics, the use of the requisition optimization policy was an even trade-off. Therefore, the use of the requisition optimization policy will neither help nor hurt the overall effectiveness of the Directorate in terms of supply availability or backorders on hand. Its continued use depends on how many more IPG 3 backorders the Directorate is willing to accept in order to obtain better fill rates in IPG 2 requisitions.

d. Changing the requisition optimization triggers.

The original results had IPG 2 and IPG 3 requisitions automatically backordered when the requisition generated was

two times the ARQ. An analysis was conducted on the requisition optimization process by holding the IPG 3 trigger constant and varying the IPG 2, and then by holding the IPG 2 trigger constant and varying the IPG 3. The results are provided at figures 8 and 9, and indicate that the original triggers produced the best supply availability and fewest backorders.

e. Changing the requisition optimization thresholds.

The original results had the requisition optimization policy implemented when an items stock level reached 30 days. Figure 10 shows the results when the policy was implemented at 15 and 45 day levels respectively. The results indicate that triggering the policy at 45 days provides the best supply availability and produces the fewest number of backorders.

FIGURE 1

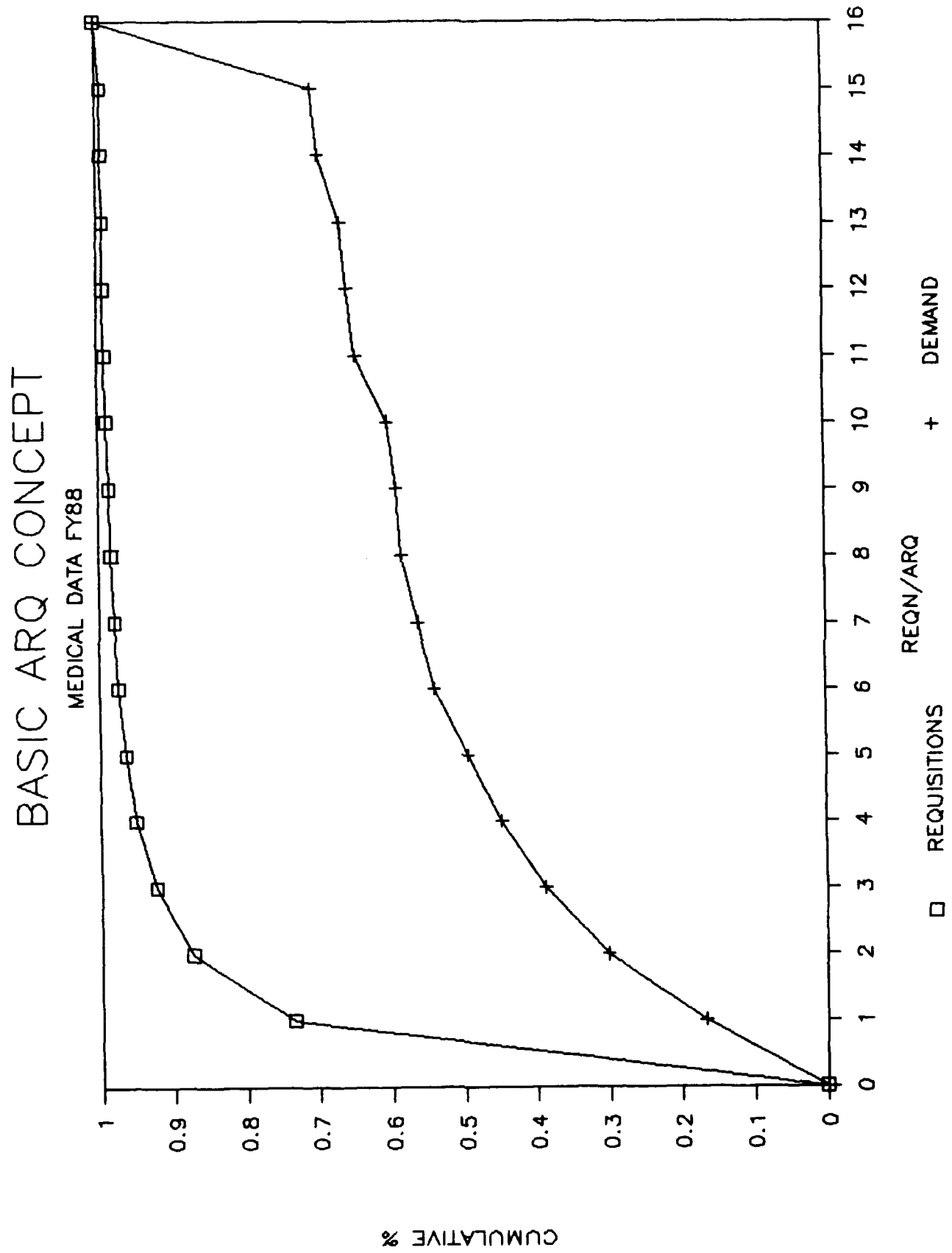


FIGURE 2

BASIC SAMMS/REQOPT RUN TRIGGERS: 2 IPG2 AND 2 IPG3

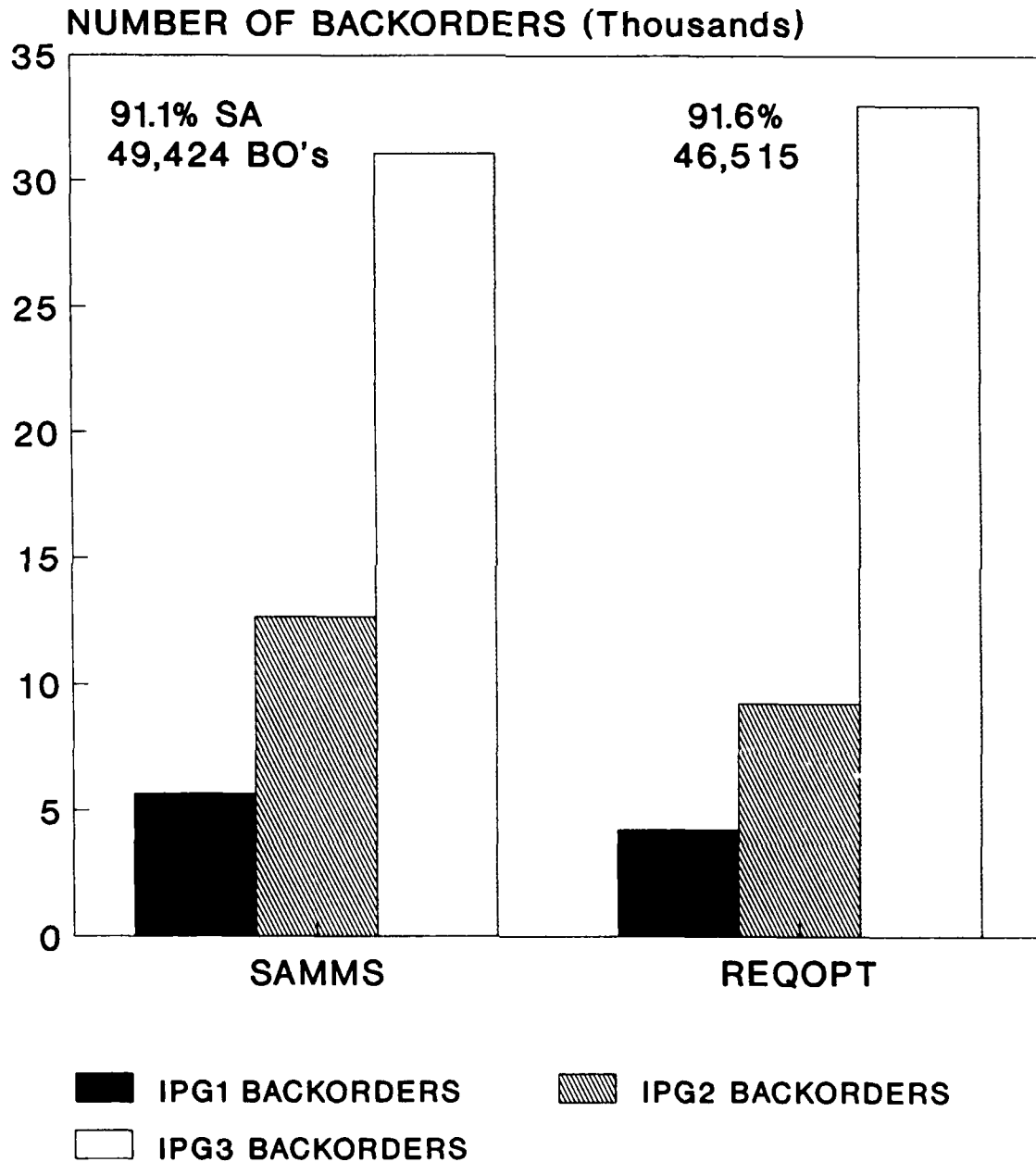
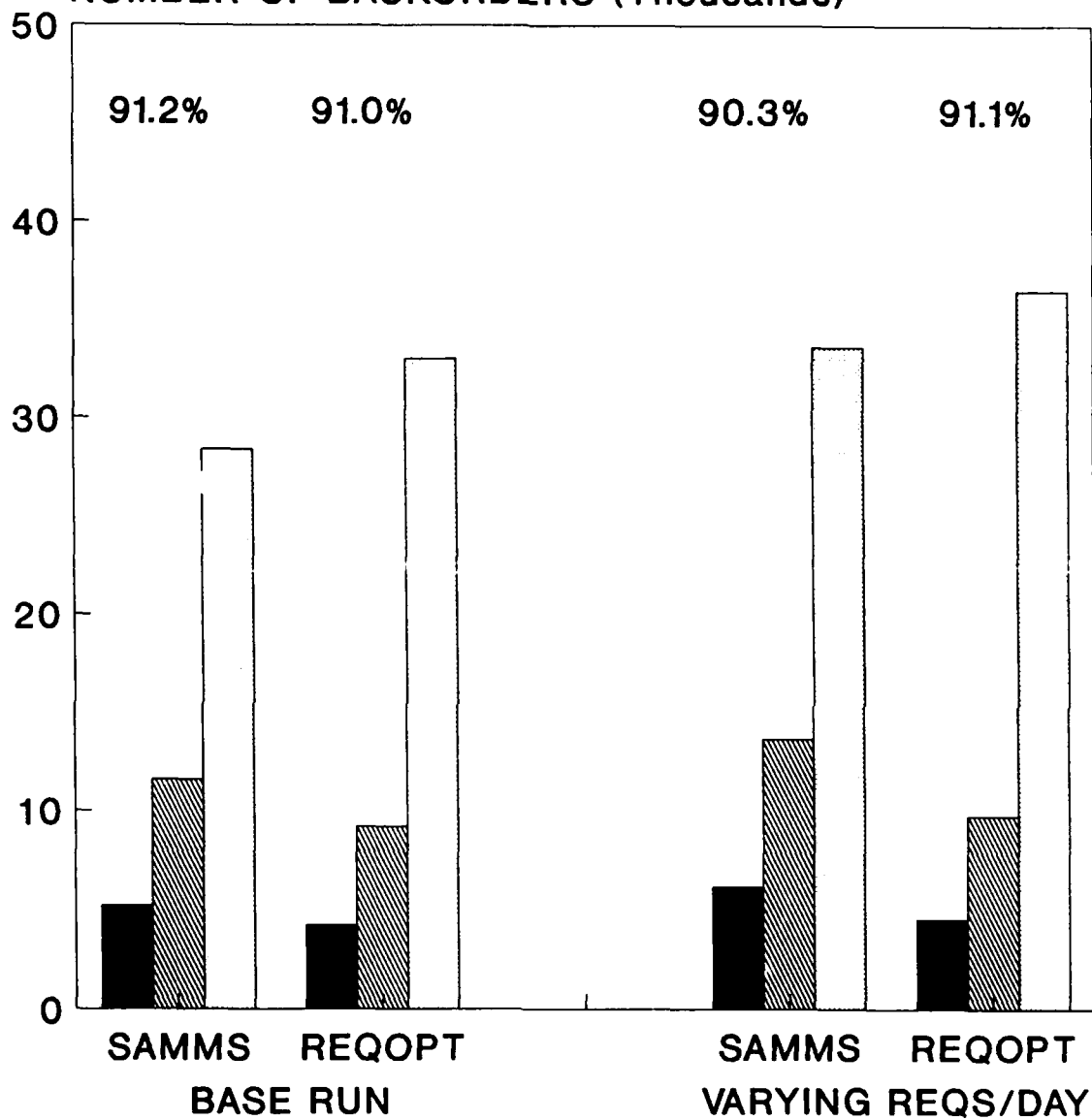


FIGURE 3

SENSITIVITY ANALYSIS OF MEDICAL DATA

SAMMS BASE RUN VS. REQOPT VARYING REQS/DAY

NUMBER OF BACKORDERS (Thousands)



■ IPG1 BACKORDERS
□ IPG3 BACKORDERS

▨ IPG2 BACKORDERS

2 IPG2
2 IPG3

FIGURE 4

SENSITIVITY ANALYSIS OF MEDICAL DATA FSC REQOPT/SAMMS RUN

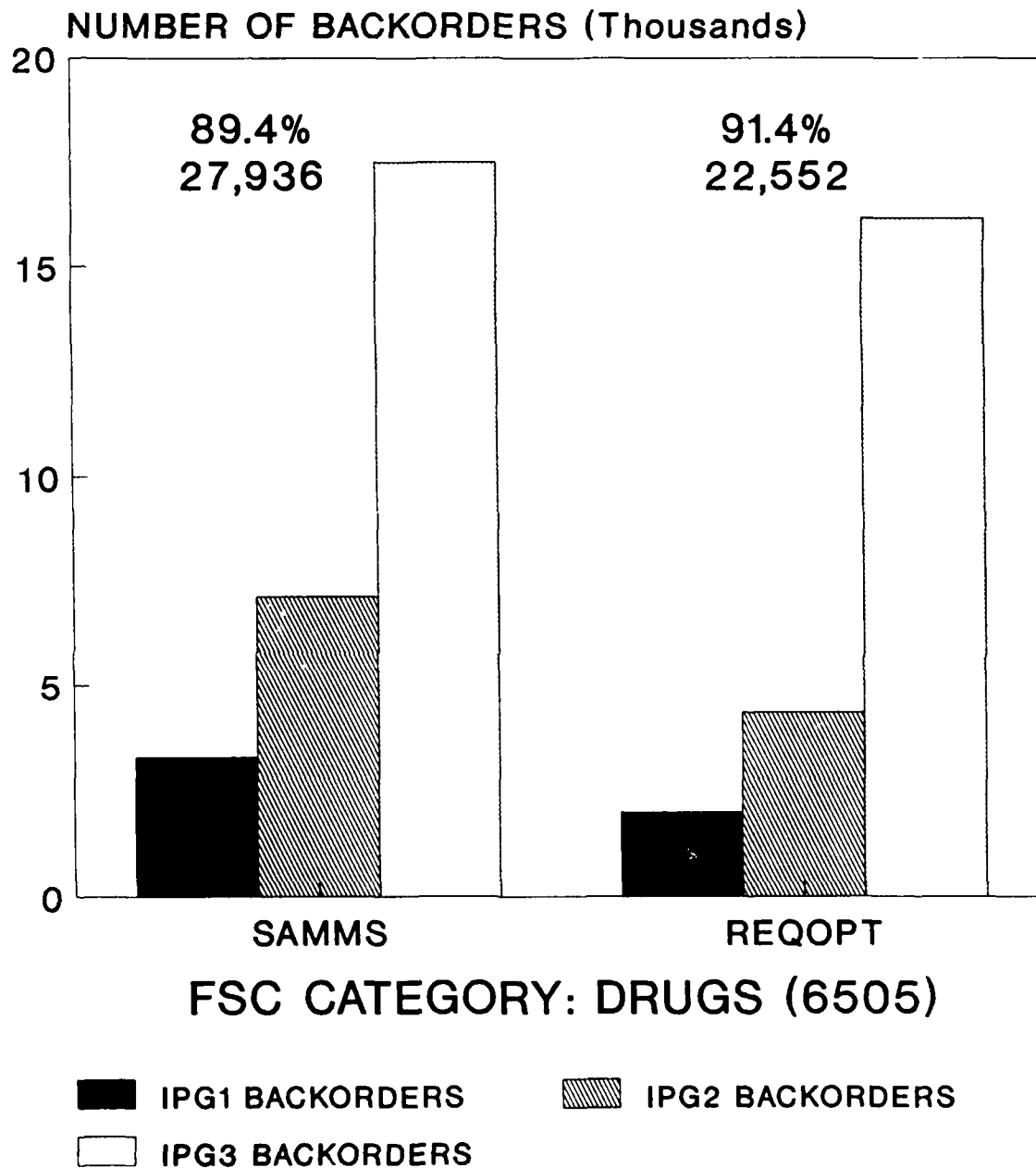
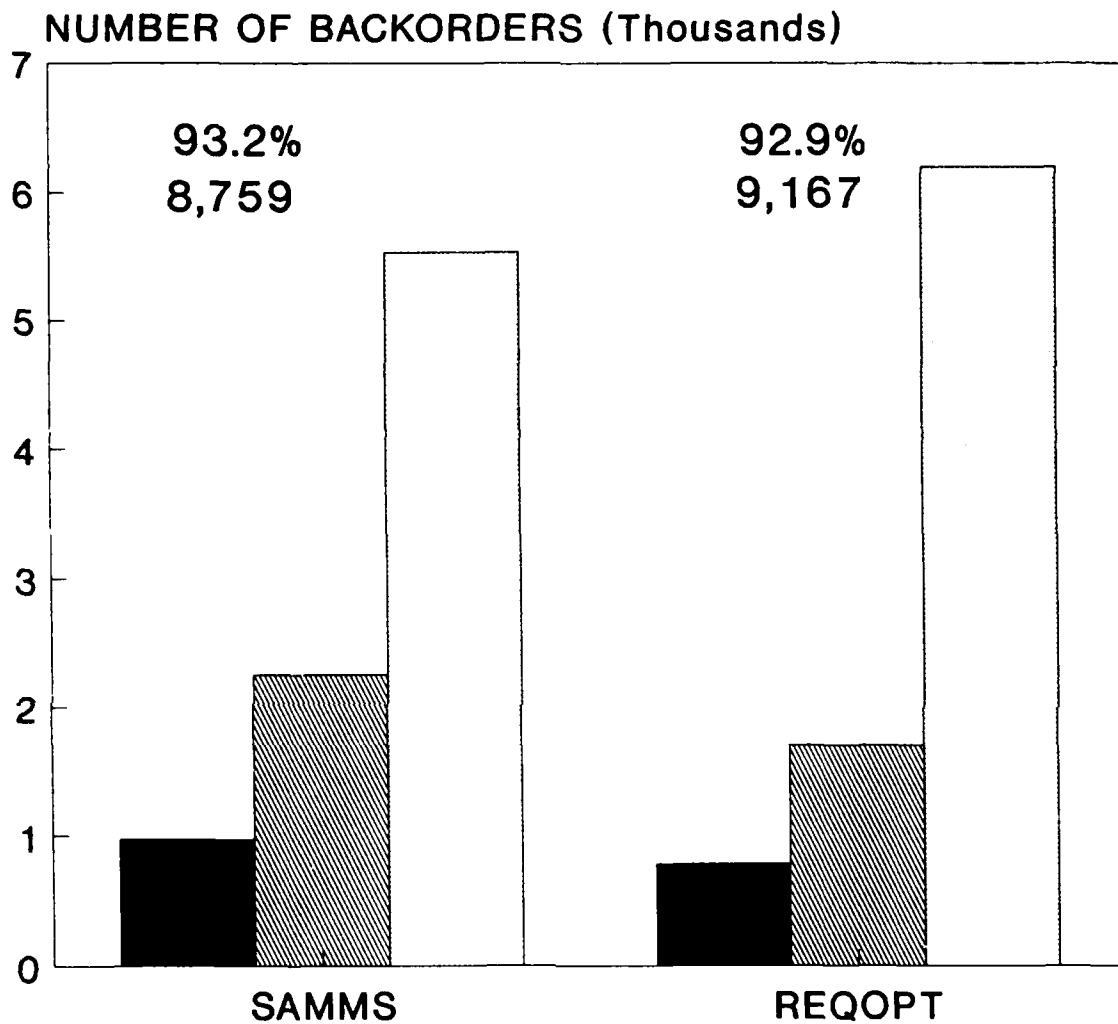


FIGURE 5

SENSITIVITY ANALYSIS OF MEDICAL DATA FSC REQOPT/SAMMS RUN



FSC CATEGORY: SURGICAL EQUIP (6510,6515)

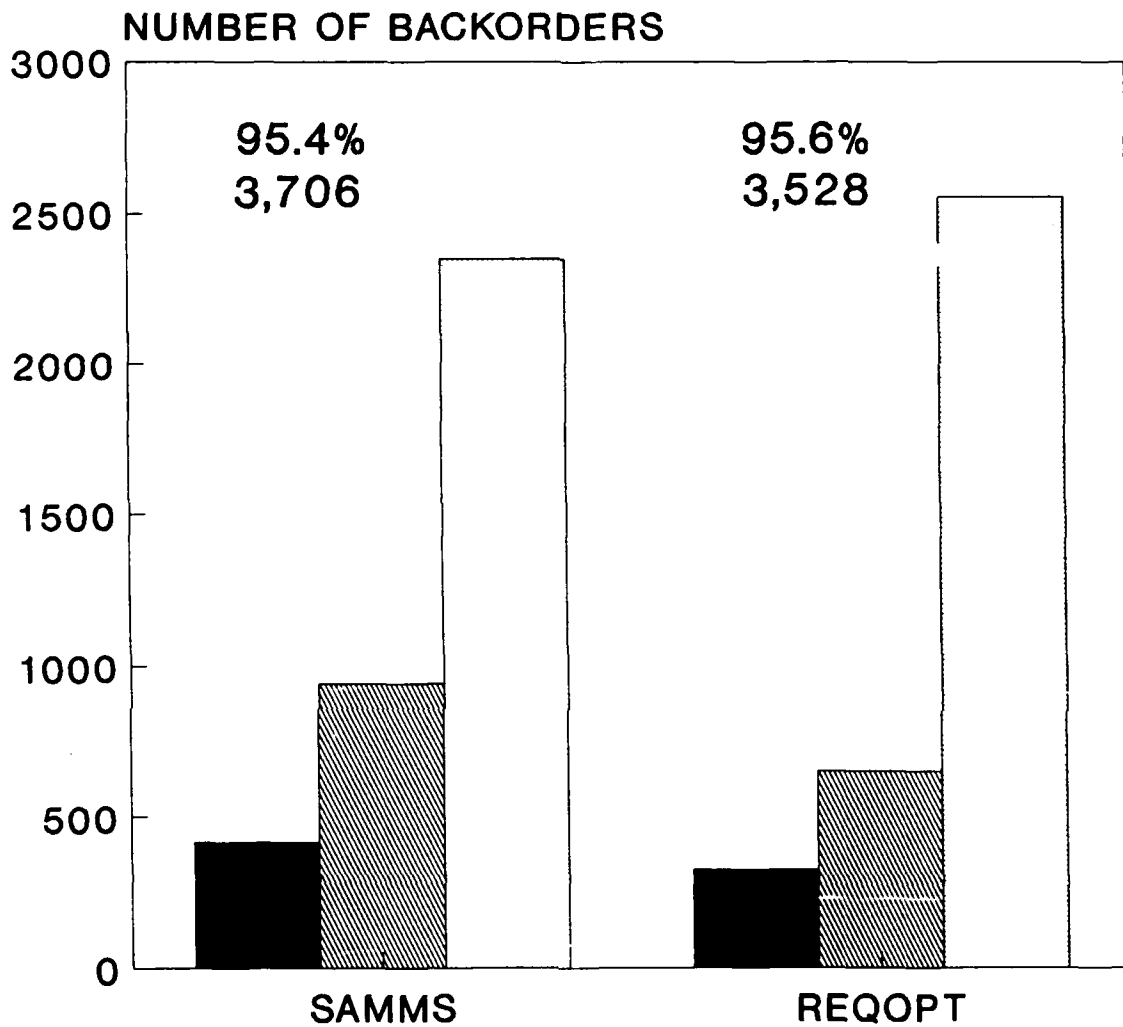
■ IPG1 BACKORDERS

▨ IPG2 BACKORDERS

□ IPG3 BACKORDERS

FIGURE 6

SENSITIVITY ANALYSIS OF MEDICAL DATA FSC REQOPT/SAMMS RUN

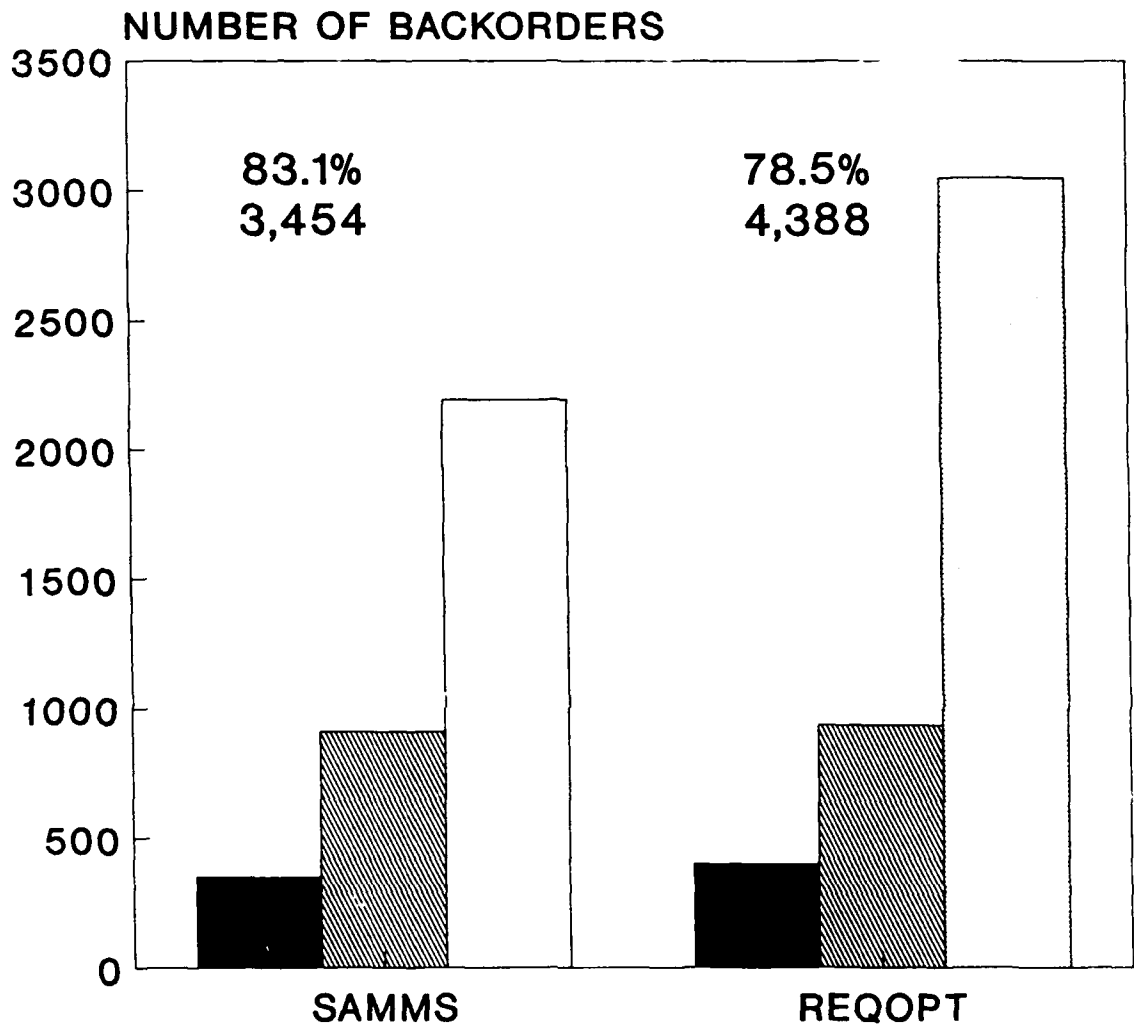


FSC CAT: X-RAY/DENTAL/HOSPITAL EQUIP
(6525,6520,6530)

■ IPG1 BACKORDERS ▨ IPG2 BACKORDERS
□ IPG3 BACKORDERS

FIGURE 7

SENSITIVITY ANALYSIS OF MEDICAL DATA FSC REQOPT/SAMMS RUN



FSC CATEGORY: OPTICAL EQUIP/LENSES
(6540)



IPG1 BACKORDERS



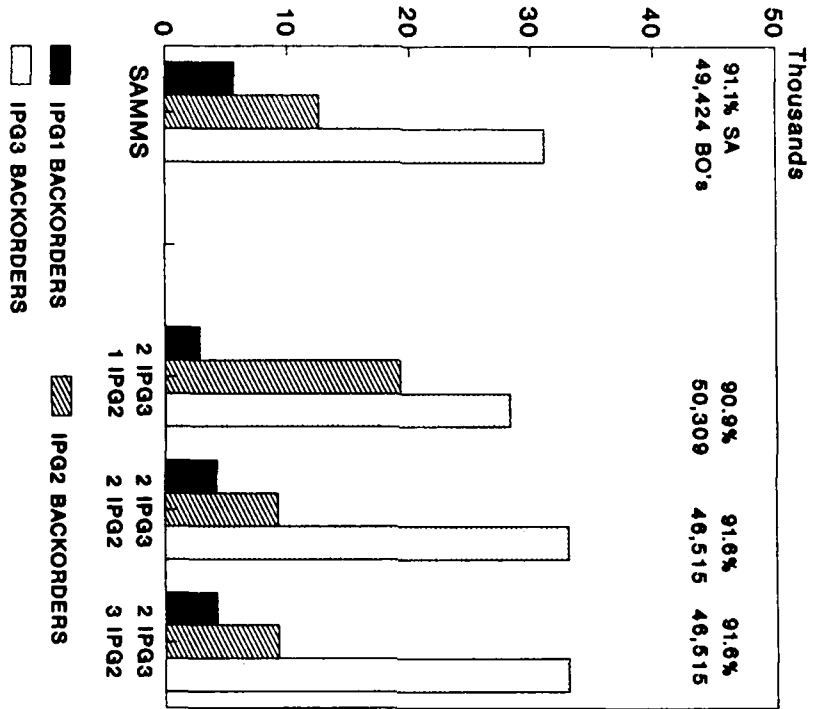
IPG2 BACKORDERS



IPG3 BACKORDERS

FIGURE 8

SENSITIVITY ANALYSIS OF MEDICAL DATA
SAMMS RUN V. VARYING RECOPT IPG2 TRIGGER



SENSITIVITY ANALYSIS OF MEDICAL DATA
SAMMS RUN V. VARYING RECOPT IPG2 TRIGGER

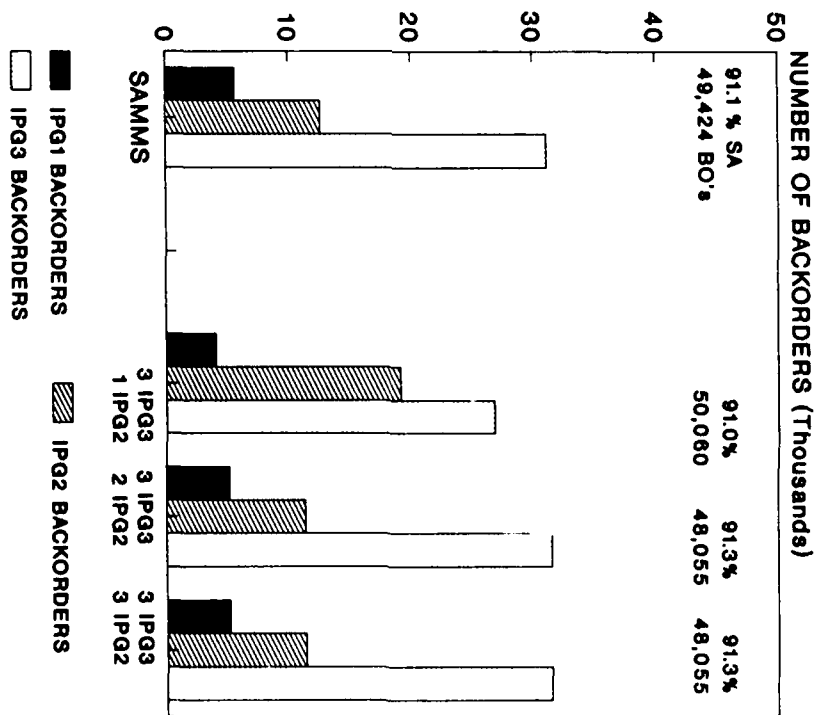


FIGURE 9

SENSITIVITY ANALYSIS OF MEDICAL DATA SAMMS RUN V. VARYING REQOPT IPG3 TRIGGER

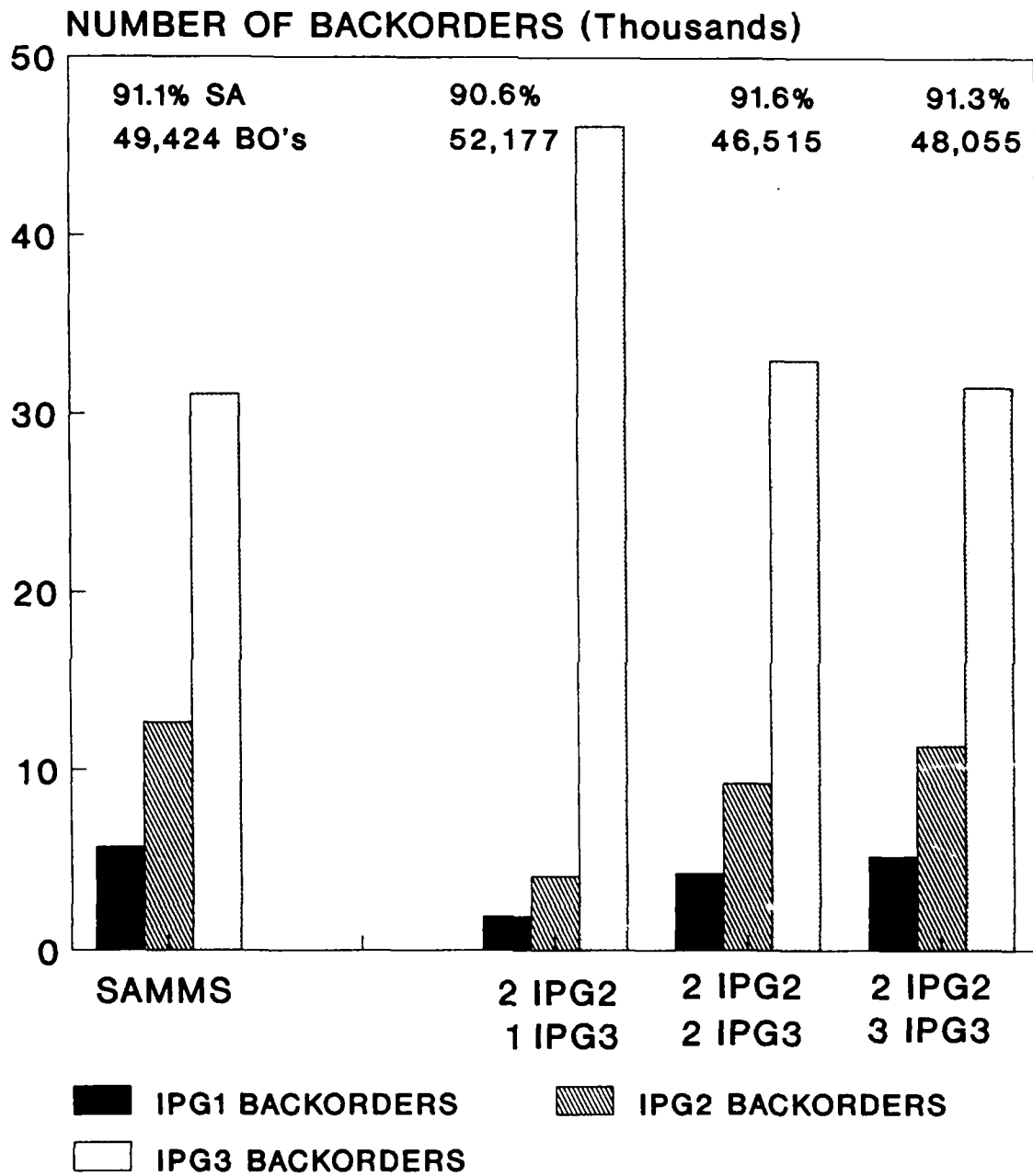


FIGURE 10

SENSITIVITY ANALYSIS OF MEDICAL DATA BASE SAMMS RUN V. REQOPT TRIGGER CHG

